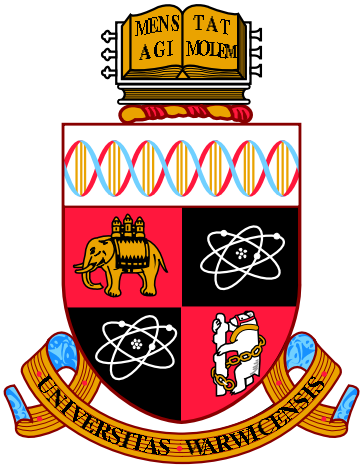


Computational Exploration of Antiviral Strategies Against Alphaviruses and Flaviviruses: A Metabolic Modelling Study

by

Sean Aller

Thesis

Submitted to the University of Warwick

for the degree of

Doctor of Philosophy

School of Life Sciences

May 2018

Contents

List of Tables

List of Figures

Acknowledgments

First of all, I would like to thank the University of Warwick and the DSTL for 3.5 years of funding. I would like to thank my supervisor Professor Orkun Soyer (University of Warwick). I would also like to thank Andrew Scott (DSTL) and Mitali Sarkar-Tyson (DSTL / University of Western Australia) for all the discussions and support they have provided as I have progressed through my PhD.

Secondly, I would like to thank the fantastic teaching team that I have worked with at the University of Warwick. They have made teaching an absolute joy and have inspired me to pursue this avenue as I further my career. In particular, I would like to give a special mention to Dr. Leanne Williams. Not only has she supported a vast number of undergraduate students as they progressed through their degrees, but she has been a source of both professional and personal help to myself. I truly am grateful to have met and worked with such a wonderful and inspiring person.

Thirdly, I would like to thank members of the OSS Lab research group for their continued support and unforgettable discussions. In particular, I would like to thank Dr. Kalesh Sasidharan for all of his helpful advice and long conversations that aided me so much in the beginning of my studies. I'd also like to thank Andrea Martinez-Veron, both for her academic support but also for providing a friendly face in the lab.

Most importantly, I want to thank my family. My family has come to include my friends, and my partner. Whilst they may not have always given me the advice I wanted, they most certainly gave me the advice I needed. It is these people, that supported me at every hour of the day, that comforted me in the extremely difficult and desperate times, and provided a sofa to sleep on, money to get me by or just simply a place to sit in silence and not have to think, that I am truly and forever grateful for. I would name them all, but there is no need: they all instinctively know who they are. I hope I have made them proud and they have enjoyed this journey with me.

Finally, I would like to mention the tragic loss of one of my friends and OSS lab member, Xue Jiang. Xue had been my friend, had at one point been my roommate, but most of all she had been a kind and caring person with fantastic scientific knowledge and skills. She has left a long lasting impact not only in my life, but in the lives of the people that knew her.

Declarations

I hereby declare that this thesis entitled "Computational Exploration of Antiviral Strategies Against Alphaviruses and Flaviviruses: A Metabolic Modelling Study" is an original work and has not been submitted for a degree or diploma or other qualification at any other university.

Chapter 1 introduces the main objectives and background for this study. This chapter provides information obtained from the literature as referenced to.

Chapter 2 explains the particular methodology I created in order to computationally model a virus as a metabolic equation. The effect of the virus introduction to a metabolic system, designated as the host, is explored. Chapter 2 is based upon the following published manuscript:

- Aller, S., Scott, A., Sarkar-Tyson, M. and Soyer, O. S. Integrated human-virus metabolic stoichiometric modelling predicts host-based antiviral targets against Chikungunya, Dengue and Zika viruses *Journal of The Royal Society Interface* **15, 20180125** (2018).

Chapter 3 investigates the prediction of antiviral targets, using the methodology developed in Chapter 2, for single-reactions within the host metabolic network. These are categorised as knockouts or host-derived enforcements. Chapter 3 is based upon the following published manuscript:

- Aller, S., Scott, A., Sarkar-Tyson, M. and Soyer, O. S. Integrated human-virus metabolic stoichiometric modelling predicts host-based antiviral targets against Chikungunya, Dengue and Zika viruses. *Journal of The Royal Society Interface* **15, 20180125** (2018).

Chapter 4 builds upon the analysis in Chapter 3, but expands to double-reaction perturbations. Chapter 4 is based upon the forthcoming manuscript:

- "Prediction of novel reaction pairs in host metabolism as virus-specific antiviral targets.", Sean Aller, and Orkun S. Soyer. (2018).

Chapter 5 is the conclusion of the PhD thesis. It concludes that the results generated from the metabolic modelling simulations and predicted antiviral targets. The theory of stoichiometric mismatch and amino acid usage, in relation to developing antiviral targets, is also discussed and used as a rational for future work directions.

Abstract

Current and reoccurring viral epidemic outbreaks such as those caused by Zika virus illustrate the need for rapid development of antivirals. Such development would be facilitated by computational approaches that can provide experimentally testable predictions for possible antiviral strategies. To this end, here the focus is on the fact that viruses are directly dependent on their host metabolism for reproduction. This thesis develops a set of stoichiometric, genome-scale metabolic models that integrates human macrophage cell metabolism with the biochemical demands arising from virus production and use it to virus impact on host metabolism and vice versa. While this approach applies to any host-virus pair, this project focuses on first applying it to currently epidemic viruses: Chikungunya, Dengue and Zika.

Overall, it is found that each of these viruses causes specific alterations in the host metabolic flux towards fulfilling their biochemical demands as predicted by their genome and capsid structure. It is predicted that all three viruses utilise the host metabolic network in a different manner than that of the host, upregulating the areas of the network that are associated with the biosynthesis of their biomass components whilst downregulating areas that are not. Subsequent analysis of this integrated model allows the prediction a set of host reactions, which when constrained inhibit virus production. These prediction recovers most of the known targets of existing metabolism-orientated antiviral drugs while highlighting a set of hitherto unexplored reactions with either broad or virus-specific antiviral potential.

To further probe how these reactions can be perturbed to inhibit virus production, the methodology created in this thesis is expanded from single-reactions to combinations of pairs of reactions, known as the double-reaction analysis. These predictions expand the novel repertoire of antiviral targets against Chikungunya, Dengue and Zika virus, and are combined with candidate drug compounds (identified from online databases) for experimental validation and implementation.

This project demonstrates that, with a combination of the application of flux balance analysis and development of a novel, integrated computational platform, the viral infection process can be better understood and potential antiviral targets predicted. The insights gained are of significant relevance, not only to the understanding of Chikungunya, Dengue and Zika viruses, but also to the overall antiviral development biotechnology cycle.

Abbreviations

Abbreviation	Meaning	Page
CDC	Centre for Disease Control	1
SARS	Severe acute respiratory syndrome	1
ECDC	European Centre for Disease Control	1
WHO	World Health Organisation	1
GSMs	Genome-Scale Metabolic Models	15
FBA	Flux Balance Analysis	18
gDW	Gram of dry-weight biomass	20
pFBA	Parsimonnious Flux Balance Analysis	22
lFBA	Loopless Flux Balane Analysis	23
dFBA	Dynamic Flux Balance Analysis	24
FVA	Flux Variability Analysis	25
COBRA	Constraint-Based Reconstruction and Analysis	27
MFA	Metabolic Flux Analysis	28
CHIKV	Chikungunya virus	32
DENV	Dengue virus	32
ZIKV	Zika Virus	32
MO	Macrophage	33
VBOF	Virus biomass objective functions	33
iHVM	Integrated host-virus metabolic model	33
+ssRNA	Positive single-stranded RNA	35
-ssRNA	Negative single-stranded RNA	35
CAC	Citric Acid Cycle	65

CHAPTER 1

Introduction and Background

1.1 SYNOPSIS

Viruses are intracellular parasites that are ubiquitous around the globe and can infect a variety of organisms, from humans to plants and insects. They are responsible for some of the deadliest disease outbreaks in human history, such as the Spanish Influenza, Ebola or Severe acute respiratory syndrome (SARS) coronavirus outbreaks. The geographical distribution of viruses coupled with their disease-causing potential increase the globally recognised threat of the introduction of new "novel" or adapted viruses into the human population, otherwise known as emergent viruses.

Several international organisations, such as the Centre for Disease Control (CDC), European Centre for Disease Control (ECDC) and World Health Organisation (WHO) have highlighted the threat of emergent and novel viruses to global biosecurity. While there has been significant progress concerning vaccine and antiviral development for some endemic human viruses, such as HIV and Measles, there remains a significant lack of possible antiviral strategies and therapeutics that are useful against these viral outbreaks.

As has been seen with the recent Ebola and Zika virus outbreaks, globally we currently lack antivirals that can be used during such an event. While vaccines confer long-term protection, antiviral therapeutics would provide a first-line defence in combatting the biosecurity risks of an outbreak in combination with other healthcare initiatives (i.e. vaccination programs, quarantine).

A significant issue in the development of such antivirals, in response to emergent

and novel viruses, is the time required for identification, experimentation and clinical verification. This is further compounded by the potential lack of information regarding the virus, its cell and tissue tropism, and the ideal conditions required to cultivate the virus *in vitro* successfully. To be able to combat the ever-growing threat of emerging and novel virus outbreaks, a rapid, easily deployable response is required before, during and after an outbreak.

In an attempt to provide this, this study attempts to fully realise a systems biology framework that utilises computational modelling of viral-mediated effects on their host's metabolic system to predict antiviral targets. This approach requires relatively few data sources, such as genomic and proteomic sequences and information regarding the virus structure and symmetry. The predicted outcomes of such an approach can then be used in experimental studies, increasing the speed of which a response against current or existing outbreaks can be generated.

1.2 VIRUS INFECTIONS

Virus infections represent not only a risk to individual human health, but also to global health and biosecurity (Cecchine & Moore 2006, Smith et al. 2014). Over the past several decades, incidences of infectious diseases of humans have increased (Smith et al. 2014), with a significant proportion of those attributed to virus infections (Fig. 1.1). In particular, there has been an observed increase in the number of RNA virus infections cases and virus of a zoonotic origin (Wolfe et al. 2007, Dunn et al. 2010, Smith et al. 2014).

The spread of these viruses are not just limited to isolated geographical regions; with increased globalisation (Smith & Guégan 2010) and a changing climate (Guernier et al. 2004, Bonds et al. 2012), virus infections are a global health issue. The understanding of what a virus "is" and how it interacts with humans in both health and otherwise, is required to protect global biosecurity and health.

1.2.1 Defining a Virus

A virus is an intracellular obligate parasite, an infectious agent that is able to enter, replicate and escape from a host cell that it infects. Viruses are composed of amino acids, nucleotides and lipids (if they are enveloped viruses). The virus structure, known as the capsid, contains the virus genetic information in the form

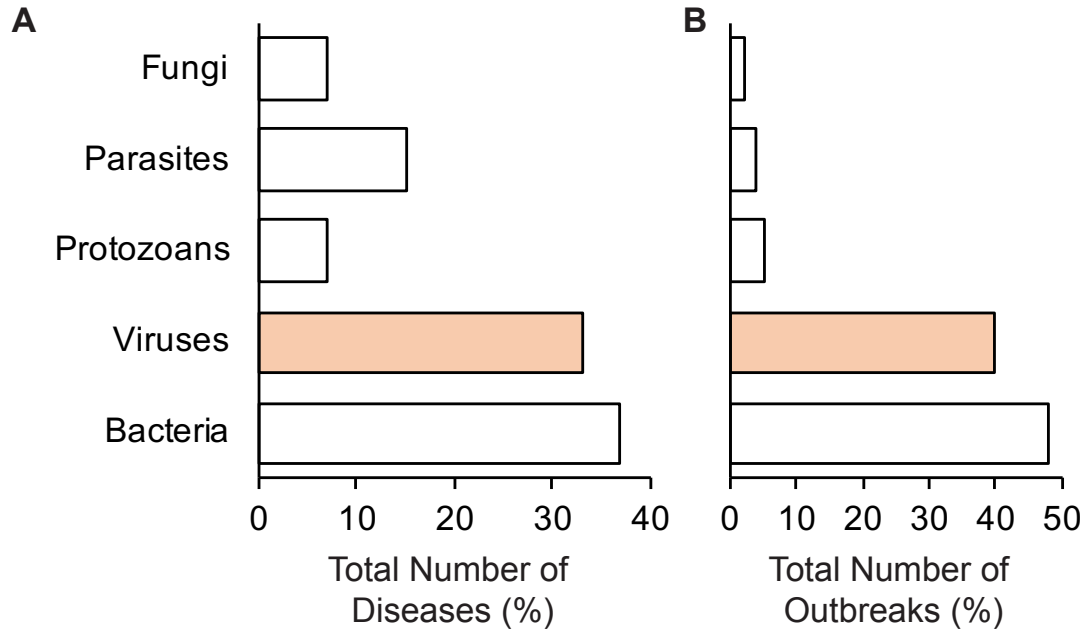


Figure 1.1: Incidences of recorded global (**a**) diseases and (**b**) outbreaks, grouped by taxonomic groups. The virus taxonomic group is highlighted in orange. Data obtained from (Smith et al. 2014).

of a DNA or RNA genome, which may be single-stranded, double-stranded or segmented.

Viruses do not have their own metabolic machinery or resources, although some viruses do carry their own genome replication enzymes, and so are entirely dependent on the host that they infect. Infected hosts are utilised by the viral agent purely for the purposes of replication and subsequent escape, to spread and infect more of the host's cells. This parasitic relationship between host and virus, and the specific circumstances that occur during the process of viral infection, may cause damage to the host cell and wider organism through diversion of crucial metabolites and compounds to the viral replication agenda (Maynard et al. 2010, Ritter et al. 2010). Virus and host are considered entangled due to this relationship, which ultimately links the virus to its host's metabolic network. This is repeatedly seen in experimental studies, where viral manipulation of the host metabolic system results in observable metabolite resource rearrangements (Goodwin et al. 2015). Furthermore, this is highlighted by observed variations in virus production levels correlating with cell-to-cell variance in growth rate and phase (Zhu et al. 2009), as well as virus infection leading to changes in host metabolism (Yu et al. 2011*a*). In particular, virus infection leads to significant metabolic alterations in the host,

in some cases resulting in up to 3-fold increase in glycolysis rates (El-Bacha et al. 2004, Jain & Srivastava 2009, Yu et al. 2011*a*) and changes in ATP production rates (Zhu et al. 2009). These observations can be seen as an emergent property of the combined host-virus metabolic system and could be related to changes in host cellular demands arising from viral production (Molenaar et al. 2009, Weiße et al. 2015). More specifically, alterations in host metabolism upon infection can be understood as viruses actively manipulating the host system to their advantage (Maynard et al. 2010), or the additional draw of metabolic components for viral production simply resulting in a re-arrangement of host metabolic fluxes.

Understanding virus infections in this manner, where alterations in metabolism are driven by the viral disease, opens up the possibility of controlling viral infection through alterations of the host metabolism (Kotzamanis et al. 2015, Maynard et al. 2010, Jain & Srivastava 2009, Ikeda & Kato 2007). Whilst this may be possible for any virus and any host organism, this study primarily focuses on the applications to viruses that represent significant risks to global biosecurity, with a focus on mosquito-borne diseases due to their global emergence and resurgence in recent decades (Gubler 2002).

1.2.2 Classification of Viruses

Viruses are classified using two main systems: the International Code for Virus Classification and Nomenclature (Adams et al. 2013) and the Baltimore classification groupings (Yu et al. 2013, Hulo et al. 2011). In the case of the latter, information regarding the viral genome composition (i.e. DNA, RNA, single-stranded or double-stranded) is used in conjunction with information on the replication intermediates to determine a broad, group-based classification system. A summary of the Baltimore classification system is shown in Fig. 1.2.

This classification system aids both in deciding the direction of viral research, for instance RNA vs DNA antiviral targets, but also in the system biology modelling efforts; virus *in silico* modelling requires the knowledge of multiple parameters, including the composition of the viral genome.

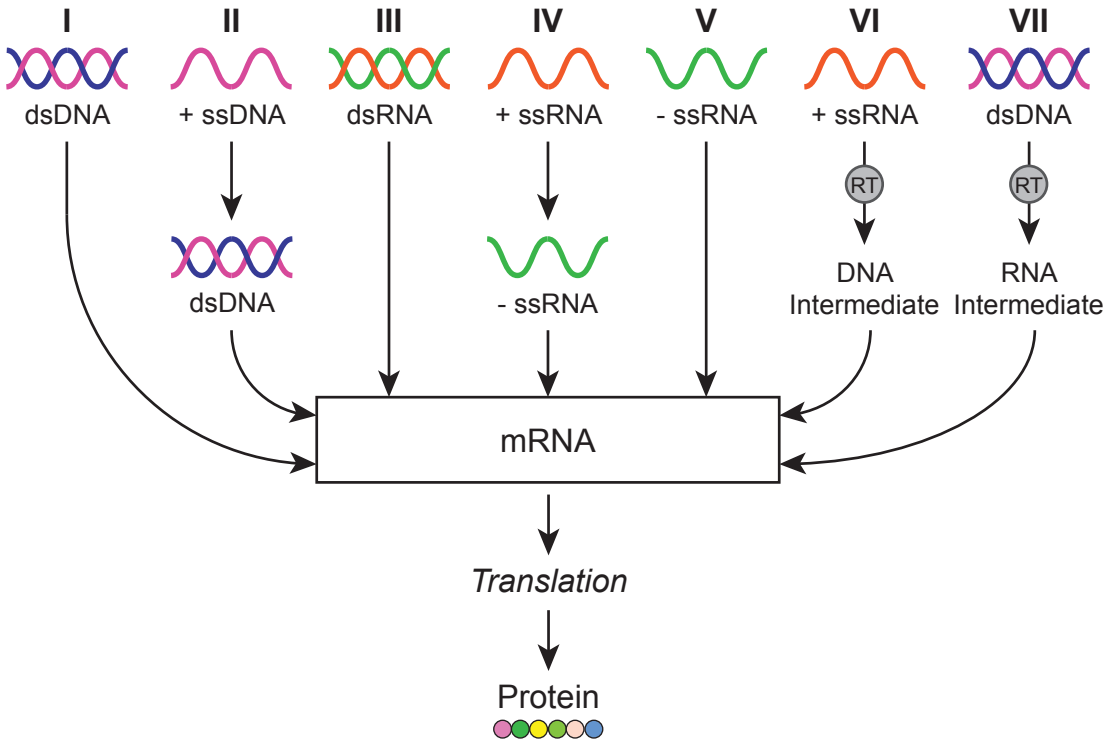


Figure 1.2: Baltimore classification of viruses. Viruses are classified based upon the method of messenger RNA (mRNA) synthesis, and their genome, into one of seven groups (I-VII): group I, double-stranded DNA (dsDNA); group II, positive-sense single-stranded DNA (+ ssDNA); group III, double-stranded RNA (dsRNA); group VI, positive-sense single-stranded RNA (+ ssRNA); group V, negative single-stranded RNA (- ssRNA); group VI, positive-sense single-stranded RNA reverse-transcriptase (+ ssRNA-RT); group VII, double-stranded DNA reverse-transcriptase (dsDNA-RT). Viruses that replicate using reverse transcriptase are indicated with RT.

1.3 EMERGING AND NOVEL VIRUS INFECTIONS

As discussed, viral diseases pose a credible risk to global human health and biosecurity (Howard & Fletcher 2012) and in particular it is viruses that are considered emerging [novel] or exotic that represent the most significant burden (Geisbert & Jahrling 2004). However, for many of these emerging viral infections antiviral discovery remains a significant challenge to overcome (Littler & Oberg 2005), and coupled with an already small set of viable antiviral drugs, further compounds the human health threat (Fauci et al. 2005). Viruses can emerge and re-emerge from many different sources, although animals and insects still remain a significant resevior for virus outbreaks, and are typically distributed over a wide geographical

area (Bonds et al. 2012, Guernier et al. 2004, Wolfe et al. 2007).

1.3.1 *Alphaviruses* and *Flaviviruses*

As discussed, the incidences of both vector-borne and viral disease outbreaks have increased between 1980 and 2010 (Smith et al. 2014). Of these viral vectors, the mosquito is responsible for the transmission of many different pathogens including many viruses. As mosquitos are arthropods, these viruses are sometimes classified as the group *Arbovirus*, and their continuing emergence presents a significant global health problem (Lwande et al. 2015, Gubler 2002, Gautam et al. 2017). More specifically, the most common *Arboviruses* transmitted by the *Aedes aegypti* mosquitos (Woolhouse et al. 2016) belong to the *Togaviridae* and *Flaviviridae* families. Both families are categorised, via the Baltimore classification scheme, as group IV viruses: they have a positive-sense, single-stranded RNA (+ssRNA) genome and their replication involve the use of a negative ssRNA (-ssRNA) genome intermediate (Yu et al. 2013). As their genomes are positive-sense, these viral genomes can be directly translated by host ribosomes to produce virus proteins. Both families are further categorised into genera, see Table 1.1.

Table 1.1: Virus classification for *Arboviruses* at the family and genera taxonomic levels. Notable species, defined as potentially effecting human health and disease incidence, are stated (Fauci et al. 2005, Woolhouse et al. 2014).

Family	Genera	Notable Species
Togaviridae	*Alphavirus*	Chikungunya Virus, Semliki Forest Virus, Sindbis Virus
	Rubivirus	Rubella Virus
Flaviviridae	*Hepacivirus*	Hepatitis C Virus
	Flavivirus	Dengue Virus, West Nile Virus, Zika Virus
	Pegivirus	GB Virus C, Human Pegivirus Type 2
	Pestivirus	Classical Swine Fever Virus

Of these viruses, there is a particular focus, in this study, on representatives of the *Alphaviruses* and *Flavivirus* due to their increased incidence of disease within the human population (Lundström 1999, Figueiredo & Figueiredo 2014, Woolhouse et al. 2016); these are positive-sense single-strand RNA-viruses with rather simple physical structures (Mukhopadhyay et al. 2005, Strauss & Strauss 1994). Both groups of viruses are enveloped, spherical particles containing a single copy of their respective RNA genome. The ultrastructure, that is the individual viral capsid components and their stoichiometries, are well known for both virus genera apart from the lipid component. Lipids are difficult to identify through structural studies and therefore the stoichiometry of the lipids, and indeed the exact type of lipid, currently lack enough detail to integrate into the modelling approach presented here.

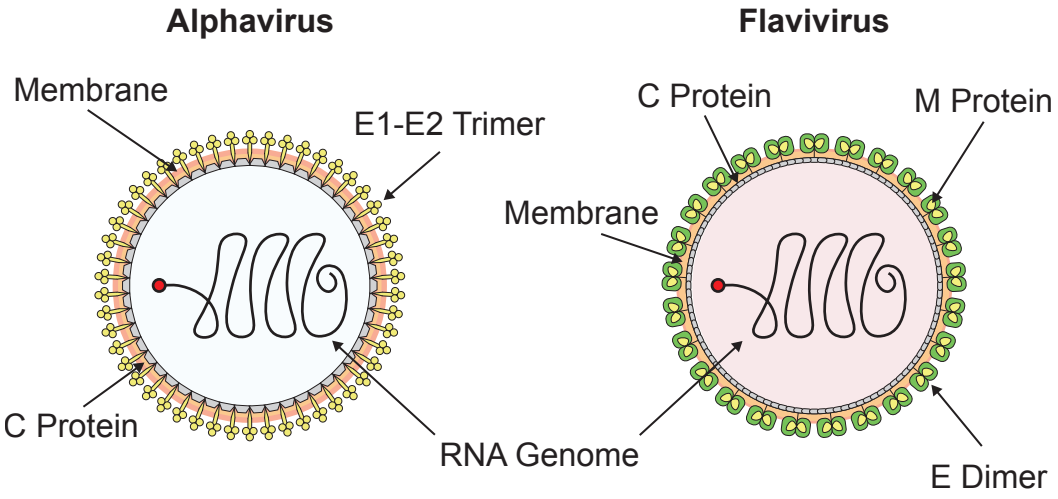


Figure 1.3: Ultrastructure diagram of *Alphavirus* and *Flavivirus* genera. Virus-associated membranes are obtained from their host's cell lipid membrane. C, capsid; E, envelope; E1, envelope 1; E2, envelope 2; M, membrane.

For a successful modelling methodology to be developed for Alphavirus and Flavivirus infections, the tissue/cell tropism of both virus genera must be considered. Viral tropism can be defined as the interactions that a virus partakes in, at various steps in its replication cycle, in order to ensure a full and successful replication cycle (Nomaguchi et al. 2012). This tropism mediates the specific host species and cell-types that a virus may infect. Viruses of both families have been observed to infect many different human cell types (Balsitis et al. 2009, Tang 2012), including monocyte-derived macrophage cell lines (Fox et al. 2015, Gollins & Porterfield 1985, Balsitis et al. 2009). Selection of a suitable host organism, and cell type, can

therefore be targeted towards human cell-types and cell-lines, with evidence that supports the use of macrophage cell models. The selection of a specific computational model and modelling technique are discussed further on in this chapter.

1.3.2 Antiviral Treatments

As highlighted through data and literature analysis, antiviral availability and discovery remains a significant global challenge. In combination with increasing emergence of viruses (Smith et al. 2014) and an extended timeline in terms of antiviral discovery and development (Littler & Oberg 2005), new technologies and methodologies must be developed and implemented to secure future biosecurity. As previously discussed, viruses are entirely dependent on their host's metabolic resources to replicate and survive. This opens up the possibility of modulating metabolism as an effective antiviral therapeutic treatment (Ikeda & Kato 2007).

Antiviral drugs are used in the treatment of viral infections, and are often used or designed against a specific virus, although broad-spectrum antivirals do exist that are effective against a wide variety of viruses (Razonable 2011, Littler & Oberg 2005). Antiviral drugs target at different points in the viral infection life cycle, and as such can be broadly categorised into four stages:

- Host entry
- Viral synthesis
- Virus assembly
- Host exit

The majority of antiviral drugs are designed so that they target specific viral proteins and inhibit their functioning, such as affecting the attachment and viral entry into the host cell. Under these conditions, the antiviral target should also be as dissimilar to the host proteins as possible, in order to limit cross-reacting with the host and thus limiting side effects (Clercq 2004, Endy & Yin 2000).

Additionally, some other antiviral drugs work through the inhibition of viral nucleic acid formation or replication. Replication of the genomic material, host or viral, requires the generation and polymerisation of nucleoside triophoshates (either ribose or deoxyribose nucleosides depending on wether it is RNA or DNA being generated, respectively). Consequently, many of the antiviral drugs target-

ing viral genome replication and synthesis are nucleoside or nucleotide analogues; altered nucleic acid molecules that interfere with the synthesis of RNA or DNA, usually through causing termination of the nucleotide sequence. In this case, the antiviral targets can be considered as "metabolic", due to using specifically altered metabolites to interfere with viral functioning. Some progress has also been made on putative antiviral targets that are the catalytic enzymes, rather than the individual metabolite substrates and products, to disrupt the production of key viral metabolites which is usually focused to the nucleoside producing reaction pathways (Marquez et al. 1988).

Previous experimental studies have highlighted the potential for metabolic components (i.e. reactions) to be a viable therapeutic and drug targets (Munger et al. 2008), and that disturbing the metabolic link between host and pathogen can result in a perturbation of that pathogen (Heinken & Thiele 2015). Currently, there are few antivirals that specifically target metabolic processes, with many antivirals targeting protein-interactions or viral entry/exit (Leyssen et al. 2008). Whilst these are effective against specific viruses (and indeed specific strains) they are both costly and timely to make, and very specific; emerging virus outbreaks require more broad-acting antivirals due to the speed at which they appear and spread through a population (Fauci et al. 2005). Additionally, there is a need to develop antiviral strategies that are able to evade or subvert virus adaptation (Endy & Yin 2000). Through traditional targeting of physical virus components, the virus is able to mutate resulting in non-recognition by the drug (due to the specific targeting of the compound to a receptor and/or coding region). By thinking of metabolism as the viruses' "environment" one can envision how easy a species could adapt to environmental changes, as opposed to individual selection pressures. Ultimately, altering the metabolic network has the potential to be far more difficult for the virus to adapt to that than of physical, specific interactions.

Developing these antiviral therapeutics requires the development of an extensive computational approach, that is able to handle the vast array of compounds, metabolic targets, host-virus interactions and indeed virus species. Such an approach would be infeasible through a purely experimental setup, therefore this study proposes to develop an integrated systems biology framework for antiviral discovery. Systems biology is based upon the principal of integrating various data sources into mathematical abstractions, to be able to predict biological outcomes (Kitano 2002, Soyer & O'Malley 2013). Such a system, if successful, can reduce the searchable space (in terms of antivirals to be experimentally validated) from

an untestable quantity to a more manageable one. To fulfill this goal, several methodologies from systems biology must be synergised into a single developmental platform and framework, that encompasses individual viruses, metabolic networks and resulting antiviral targets.

1.4 SYSTEMS BIOLOGY AND MODELLING

1.4.1 Models and Modelling

Realising a systems biology approach to host-virus modelling, and indeed any modelling situation requires the development and implementation of models (Coscia et al. 2011, Dhurjati & Mahadevan 2008). In systems biology, models are mathematical descriptions of biological systems, for instance a set of mathematical equations that represent a reaction network. However, models are not able to describe all components of a biological system (i.e. the concentration of every metabolite within a cell and its environment), and therefore some aspects of the biological system must be simplified or excluded from the mathematical model. Therefore it is important to acknowledge what can and cannot be achieved with a mathematical model but also how to interpret the results in light of such omissions and simplifications.

There are significant advantages to computational modelling. Firstly, the creation of a model may require a much smaller amount of data or information than the corresponding biological experiment (Dhurjati & Mahadevan 2008). Secondly, if the model is constructed well and has been compared to biological and experimental data, the predictions and outputs of the model can be relatively accurate and indicative of the real-biological response. This, combined with the [usually] quick nature of computational modelling, makes this a viable approach for simulating biological systems.

Just as there are many different types of biological entities, there are multiple model types that can be utilised in systems biology. The majority of the work covered in this thesis involves the construction and analysis of network-based biological models. As such, the different modelling approaches for aforementioned networks, along with potential advantages and disadvantages, are discussed below.

Applications of Network-Based Models: Representing Metabolism.

Network-based modelling and analysis allows for complex interactions between biological entities, whether that be infectious and susceptible agents or reactions and metabolites, to be predicted and analysed and their interactions with the broader biological-environment assessed (Keeling & Eames 2005, Shlomi et al. 2008, Eubank 2005, Papin et al. 2004). The construction and subsequent methodologies that can be utilised for their analysis are shown in an overview, Fig. 1.4.

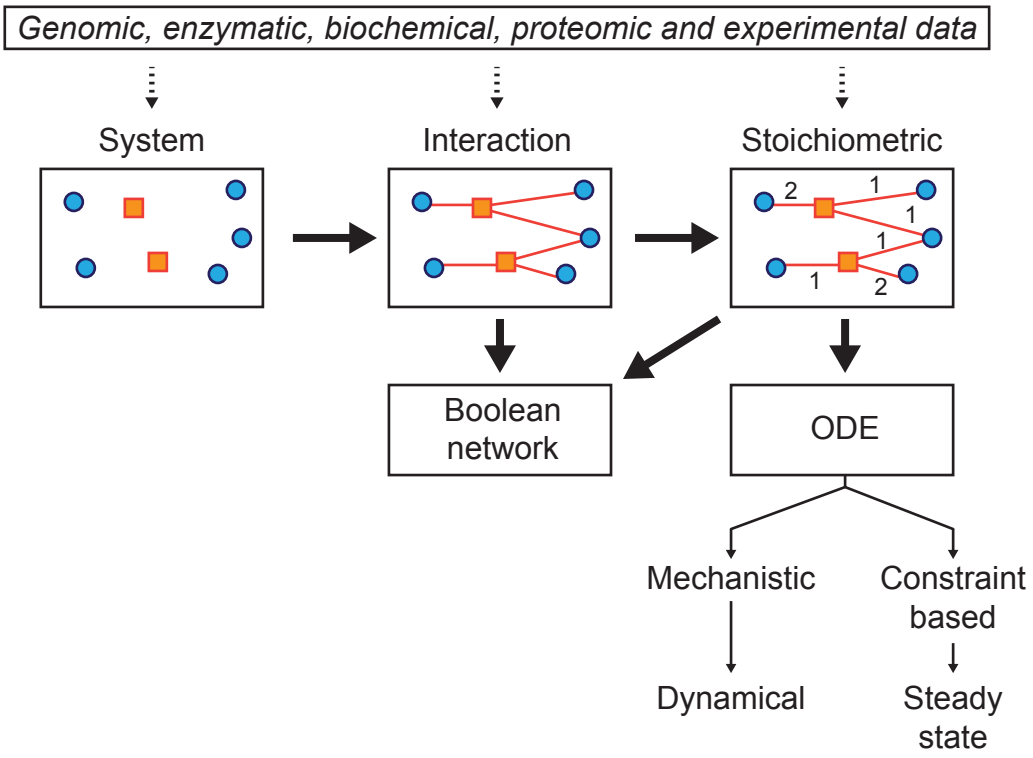


Figure 1.4: Toy schematic of how various data sources are combined to form different network-based models. Solid arrows indicate direct evolution from one model type to another. Two different groups of entities are shown with blue circles and orange squares. ODE, ordinary differential equations.

When constructing a network-based model, one can think of biological entities (i.e. metabolites and reactions) in isolation, producing a "System" model of unconnected entities (or nodes), or in a network-based view, such as "Interaction" models. These network models involve the creation of node-to-node connections, known as edges. The combination of nodes and edges forms the basis of any network-based model, and can be represent as a graph of the nodes and edges (as shown in Fig. 1.4 in the "Interaction" and "Stoichiometric" examples) or can be represented mathematically as a combination of vertices and matrices. Whilst

multiple types of network-based models can be created, the discussions presented in this thesis concern only metabolic networks. Therefore, here the different types of metabolic network-based constructions and methodologies are discussed.

Stoichiometric Models. Biochemical networks, such as that of metabolism, can be described using network-based structural models (Thiele & Palsson 2010, Stincone et al. 2015). The essential components of a metabolic network model are the metabolites [compounds] and the reactions, including transport processes, that change/alter the metabolic compound. An additional layer of information is gathered from the enzymes that catalyse reactions, and the proteins associated with the transport of compounds in the biological system. In this way, each reaction and transport process can be assigned to a unique identifier.

Stoichiometric models retain the structural information of the reaction network, where reactions and compounds are considered as separate groups of entities. Compounds usage in reactions is denoted using stoichiometric coefficients, which describe the proportions of reactants and products in a given reaction, for example:

$$1A + 2B \longrightarrow 1C \tag{1.1}$$

The reactants in this equation (Eq.1.1) are A and B and the product is C. The stoichiometric coefficient, indicated by the number preceding the compound, indicating the proportion of that compound that contributes to the reaction. For instance, describing that 1 mole of A combined with 2 moles of B yields 1 mole of C. It should be noted that stoichiometric coefficients are not unique (i.e. a coefficient of 1 can be assigned to many compounds). The reaction is denoted using the arrow, and arrow direction indicate directionality (in this case, unidirectional).

A stoichiometric matrix, required to build a mathematical metabolic model, is a compact description of such a reaction network. Stoichiometric coefficients c are combined with the metabolites m and reactions r to yield the stoichiometric matrix S. Using this notation, Eq. 1.1 can be described as

$$c_{ij}m_i + c_{ij}m_i \xrightarrow{r_j} c_{ij}m_i \tag{1.2}$$

$$\text{for } i = 1, \ldots, m \text{ and } j = 1, \ldots, r \tag{1.3}$$

and is represented in matrix format as

$$S = \begin{bmatrix} c_{ij} \\ c_{ij} \\ c_{ij} \end{bmatrix} \tag{1.4}$$

which yields the stoichiometric matrix S, in this example for this (j) reaction (r), where each row corresponds to a different (i) biochemical compound (m). The size of the stoichiometric matrix is formally defined as $m \times r$. An example for a larger reaction network, with multiple branching points, and the associated stoichiometric matrix is shown in Figure 1.5.

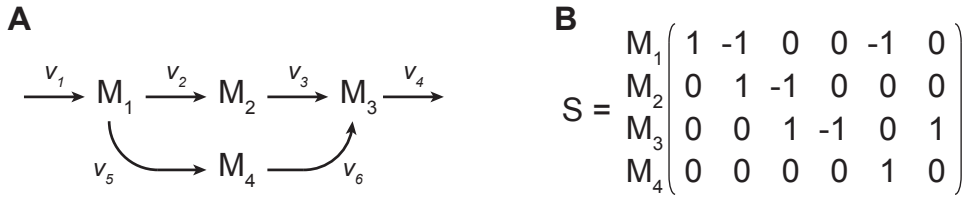


Figure 1.5: A small reaction network system and the corresponding stoichiometric matrix S. Arrows denote reactions associated with metabolites M, that run at a certain flux rate v. Stoichiometric coefficients are shown in the S matrix as: zero (reaction does not produce or consume M); positive value (reaction produces M); negative value (reaction consumes M).

The stoichiometric matrix only captures the interactions between the different biological entities, the metabolites and the reactions, and how they are connected. This information can be used to construct a network-graph, detailing the interactions between the various nodes, but is not sufficient alone to describe a complex system such as a metabolic network. To further develop and enhance stoichiometric models, further information can be applied and considered in order to expand the modelling approach.

Kinetic Models. The ability to investigate and predict changes in metabolic systems, through the use of mathematical models, remains one of the significant challenges in systems biology. In addition to stoichiometric models, kinetic models provide a means for testing responses of biological systems (i.e. metabolism) to various effectors. However, unlike stoichiometric models, kinetic models can be used for dynamic responses of a system to various perturbations such as changes in environmental conditions (i.e. glucose availability) or genetic alterations (i.e.

repressors) (Smallbone et al. 2013).

Small kinetic network models, such as a part of the central carbon metabolic pathway for *E. coli* (Jahan et al. 2016), have been created and successfully tested. However, these studies highlight some problems with the development and implementation of kinetic models. Firstly, the development of kinetic models is hindered by the lack of information regarding complex kinetic variables, such as metabolite concentration and enzyme abundance (Saa & Nielsen 2016, Steuer et al. 2006). Secondly, even with the availability of adequate information, the computational cost of running kinetic models increases exponentially as the scale of the reaction networks increase (Resat et al. 2009). These factors significantly limit the systems that can and have been successfully modelled with an accurate kinetics-based approach.

There have however been some smaller studies that focus on subsections of an organisms reaction network which have extensive levels of details regarding annotations and data availability (Jahan et al. 2016, Kerkhoven et al. 2014). There have been previous attempts at kinetic models for more complex organisms, mainly eukaryotic organisms such as humans (Bordbar et al. 2015); however these come with significant assumptions about the modelling approach utilised. Many of the kinetic and thermodynamic parameters are obtained through estimation, fitting or simulation, rather than from empirical data sources. While these may yield plausible modelling parameters, the lack of empirical data means that they cannot be compared or validated beyond the initial predictions, compounding further the issue of model accuracy and reproducibility. Overall, kinetic models provide many advantages over stoichiometric models, mainly their ability to deal with dynamic and changing variables and include many other non-network based parameters. However, the lack of information available regarding parameters and thermodynamic constraints for entire human cells, as opposed to individual reactions and pathways, makes the stoichiometric model a preferable choice over kinetic for modelling whole-cell metabolism, as used in this thesis.

1.4.2 Structural Models of Metabolism

An extension of the stoichiometric model, metabolic models are constructions of a stoichiometric matrix directly from a set of metabolic reactions. As previously described, reactions are deconstructed into the metabolites (reactants and products)

and assigned stoichiometric coefficients that denote their reaction-proportions and whether they are consumed or produced.

The scope of the metabolic model only describes the interactions between metabolic elements in a network or organism, which are then represented in the stoichiometric model format. In graph notation, a metabolic network can be considered as a bipartite graph, with metabolites forming one set of nodes and reactions forming another set (Fig. 1.6). Stoichiometric metabolic models also include additional information regarding the directionality of reactions, defined as whether a reaction can proceed in the forward and reverse direction (bidirectional) or forward direction only (unidirectional) (Beard et al. 2004). This information is usually encoded in a vector that is equal in length to the number of reaction entities, and indicates whether a reaction is bidirectional ("True": 1) or unidirectional ("False": 0).

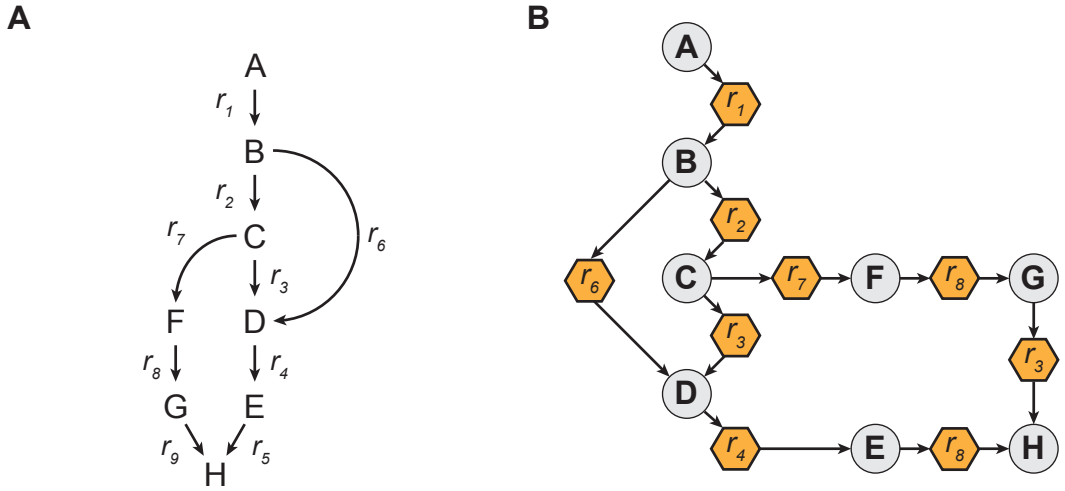


Figure 1.6: Representation of a (**A**) small reaction (r) network with 9 reactions and 8 metabolites, denoted by capital letter, and (**B**) a bipartite representation of the interactions between the metabolite (grey filled circle) and reaction (orange filled hexagon) nodes. Note that reaction directionality is conserved between both representations.

It should be noted that in this example (Fig. 1.6) only metabolites and reactions that are intracellular and are not involved with external-internal transport are shown. A subset of metabolites in the metabolic model must be supplied from an "external" source (usually designated the *in silico* media), where the supply of that metabolite is set to a particular uptake rate. For instance, in Figure 1.6A, metabolite "A" would be supplied by an external transport reaction from the *in silico* media.

Metabolic models, from the networks to the stoichiometric matrices, can be gen-

erated in many ways (Le Novère et al. 2005). Many types of metabolic model exist, from small toy or "core" networks (only composed of a few selected reactions) to the more commonly used genome-scale metabolic model (Thiele & Palsson 2010).

Genome-Scale Metabolic Models

Mathematical metabolic models derived from an organism's genome sequence are known as Genome-Scale Metabolic Models (GSMs). GSMs are one of the most commonly used metabolic models due to their relative ease of creation. They are derived from the genome of an organism and contain all the information related to annotated metabolic genes in an organism. Through gene-enzyme association, this provides a list of all possible enzymes and therefore reactions that are then translated into the model, generating a stoichiometric model from the genome. Because the model is derived from gene annotations, the model is only considered as accurate or representative as the annotations themselves; in other words, reactions cannot appear in this model that are not associated with an annotated gene. Additional reactions can be added in after derivation from the genome, to fill any sections of the network that may be missing due to lack of annotated genes or genome sequence for that region. The goal of the GSM is to simplify the metabolic components of an entire organism into a stoichiometric matrix, which can then be utilised for analysis and predictive outcomes (Henry et al. 2010).

Currently, many different GSMs exist and have been used for a variety of metabolic analysis, engineering and synthetic biology development (Papin et al. 2003, Rodrigues et al. 2013, Chang et al. 2010, Kotte et al. 2010, DeBerardinis et al. 2008). Initially, GSMs were created for so-called "model organisms". These are organisms that have been used in experimental studies numerous times, and have well-annotated genomes, making them excellent candidates for a metabolic model generation. They were, however, prokaryotic organisms, due to the difficulties in constructing eukaryotic organism metabolic models.

Development of prokaryotic and eukaryotic metabolic models

Initially, metabolic models were considered for small sections (known as "toy" or "core" models) of prokaryotic metabolism; specifically, *E. coli* was selected as the initial model organism, and the central carbon metabolism pathways (glycolysis,

the citric acid cycle, the pentose phosphate pathway) were selected for this core model. This was in part due to the wealth of information that describes *E. coli*, its biological processes, and the documentation of individual steps within central carbon metabolic pathways (Liu et al. 2014). Additionally, due to prokaryotes having a single cellular compartment, the identification and organisation of metabolites and reactions are simple; all reactions and metabolites occur in either the intracellular or extracellular "compartment" (Vellai & Vida 1999).

Eukaryotic organisms, however, are multi-compartmental, which can make it difficult when generating a metabolic model (Tanaka et al. 2006). Specifically, this makes identifying in which compartment reactions and metabolites reside, and indeed if they reside in specific compartments or are ubiquitous throughout the cellular system, difficult. There is also the issue of compartment-to-compartment transport and the identification of the relevant transport links that may exist within the cellular structure (Hatzimanikatis et al. 2005). To combat this issue, and in particular, for GSMs, transcriptomic and genomic data can be used to establish localisation of metabolites and reactions within cells (Gutierrez & Lewis 2015). This is also useful for multi-cellular eukaryotic organisms, where multiple cell types, with potentially different metabolic network topologies, exist. The transcriptomics and genomic data can be used to "prune" the genome of the organism to identify cell-specific metabolic genes, further differentiating the metabolic network to a genuinely cell-specific one.

Human Metabolic Models

Expanding upon the developments of eukaryotic model creation, human metabolic models have also been created for both the entire genome and various specific cell types (Chowdhury & Maranas 2015). Most notably, collaborative efforts with many research groups lead to the creation of the human GSM titled "Recon 1" (Duarte et al. 2007). This GSM was created utilising data from the human genome project, as well as integrating previous attempts at creating a human GSM; The Edinburgh Human Network (Ma et al. 2007). This GSM encompassed all annotated metabolic genes within the human genome, and in doing so represented the metabolic potential of every combined cell-type. However, this presents the issue that no single cell type has access to the network topology of every other cell type, requiring the development of cell-specific metabolic models. This also aids in the analysis and prediction of cell-specific experiments.

The first primary human cell-specific model was HeptaoNet1 (Gille et al. 2010), which was created by combining reactions and metabolites from both Recon 1 and The Edinburgh Human Network (Duarte et al. 2007, Ma et al. 2007). Further developments led to the creation of a metabolic reconstruction of a human macrophage (Bordbar et al. 2010), which utilised Recon 1 as a base model with several sets of clinical gene expression data to refine the metabolic model. A metabolic model for human astrocytes was also created to study the effect of neurological disorders and how the pathology of these diseases may either affect or be influenced by the astrocyte's metabolic network function (Martín-Jiménez et al. 2017). Further refinements were conducted on the original Recon model, to yield Recon 2 (Thiele et al. 2013, Swainston et al. 2016) which increased the number of metabolic components (due to increases in gene annotations) and was curated by a more extensive consortium than before. Ultimately, while there are fewer metabolic models for humans (concerning GSMs), the selection of models that are available are well curated and supported by continued research and development.

1.5 LINEAR OPTIMISATION AND FLUX BALANCE ANALYSIS

Once a metabolic model has been created, whether it be a toy, core or genome-scale model, it can be analysed using a variety of linear optimisation techniques.

1.5.1 Linear Optimisation

Linear optimisation is a mathematical method of optimising a model, for an objective, given a set of linear equality and inequalities in order to achieve an outcome for a given objective (Broyden 1994, Stroud 1990). This objective, also known as the objective-function, is a linear objective function that in combination with the equalities and inequalities is used to find a feasible solution; this can be considered as the "outcome" of a system given a set of pre-determined constraints. A formal definition of linear optimisation is described in Eq. 1.5.

$$\begin{aligned} \text{maximise} \quad & \mathbf{c}^{\intercal} x \\ \text{subject to} \quad & Nx \leq a \\ & x \geq 0 \end{aligned} \tag{1.5}$$

where x is the vector of variables that are being determined, and a and c are vectors of coefficients. The matrix N is of coefficients also. The objective function is defined as the expression that is maximised (or minimised), $\mathbf{c}^T \cdot x$ in this example.

Once a linear optimisation problem (LP) has been created, using a set of linear constraints and an objective function, the LP is then fed into an LP solver which applies an algorithm to solver the LP for the given objective. The algorithm utilised by most LP solvers is the simplex method.

Linear constraints from the LP define an LP solution space, known as the feasible space, which forms a convex polyhedron; an n-dimensional shape with vertices, edges and faces that contains a finite number of points. The simplex method calculates a solution to the LP in an iterative process (Stone & Tovey 1991).

Initially in the first phase, a starting vertex is selected, and the simplex algorithm moves along the edges from this vertex until a feasible solution is found. If a solution cannot be found within the given feasible space, then the LP result is infeasible. Upon finding a feasible solution, phase two describes the application of the simplex algorithm using the solution found in the first phase as a starting point. Again, the movement along the edges is iterated until a point is reached that any further movement along any of the edges will reduce the objective function; essentially this point is the optimum solution for the LP (Nering & Tucker 1992, Dantzig & Thapa 1997).

Linear optimisation forms the basis for the most common form of stoichiometric metabolic model analysis: flux balance analysis.

1.5.2 Flux Balance Analysis

Linear optimisation can be used for the optimisation of network flow, and therefore can be applied to the stoichiometric models of metabolism. More formally, linear optimisation of a metabolic model is known as flux balance analysis.

Flux balance analysis (FBA) is a mathematical, constraint-based modelling method used to simulate reconstructions of metabolic networks (Orth et al. 2010). FBA is computationally inexpensive; both the linear optimisation and requirements for input data (in terms of model construction) are considered less intensive than other modelling techniques, such kinetic modelling discussed in in §1.4. No information

is required regarding dyanmics, metabolite concentrations or other thermodynamic constraints. This is achieved due to two assumptions: steady-state and optimality.

The steady-state assumption considers that in a reaction mechanism (i.e. a metabolic reaction) that a compound or intermediate is consumed at the same rate as it is produced; there is a net change in concentration of 0 (Orth et al. 2010, Beard et al. 2004). When applied to an entire metabolic network, this assumption enforces that the overall net change in total rate of metabolite production and consumption is 0.

FBA describes and predicts how the flow of metabolites through the metabolic network, here called reaction "flux", is distributed optimally for a cell. This notion of optimality is achieved through multiple constraints on the metabolic flux distribution. Firstly, individual reactions (columns in the stoichiometric matrix) are constrained by a minimum and maximum flux value, known as the lower and upper bounds (respectively). The flux bounds acknowledges that enzymes have limited capacity to catalyse a reaction, and a maximum value can be set using the upper bound. The lower bound is to signify reactions that may be reversible (bidirectional) if it is a non-zero negative number, and irreversible (unidirectional) if it is zero or a non-zero positive number. The linear inequalities that arise from the setting of these flux bounds are shown in Eq.1.6.

$$lb_i \leq v_i \leq ub_i \tag{1.6}$$

where indexation i is over all reactions in the metabolic model. The lower (lb) and upper (ub) bounds are determined from experimental data or literature. If bound values are not available they are traditionally set to lower bound = 0 and upper bound = $+inf$ (Orth et al. 2010).

Secondly, the flux distribution steady-state assumes that the metabolic model [organism] has entered a state where the influx of metabolites equals the efflux; the change in a specific metabolites concentration does not change as the producing and consuming fluxes cancel each other out. This assumption ensures that all variables remain constant over time. More formally, steady-state is defined as Eq. 1.7.

$$Sv = 0 \tag{1.7}$$

where S is the stoichiometric matrix of the metabolic model, and v is the flux vector that satisfies the steady-state assumption.

Finally, an additional constraint is added to the flux distribution so that a unique optimal solution is yielded. On their own, the steady-state assumption (Eq.1.7) and flux distribution vector constraints (Eq.1.6) provide a set of feasible solutions for the metabolic model, but do not result in a unique solution; this is due to the linear optimisation problem being underdetermined as there are generally many more reactions than there are metabolites. To solve this, the metabolic model flux distribution is optimised [maximised] for a biological-relevant objective function Z (Eq.1.8).

$$Z = \mathbf{c}^{\intercal} v \tag{1.8}$$

where Z is the optimal value for the objective function $\mathbf{c}^{\intercal} v$. c is a vector of zeros with a non-zero value (usually 1) at the index position of the reaction selected as the objective function. This objective function, in most cases, is the production of biomass for replication (Orth et al. 2010, Schuster et al. 2008), which is known as the biomass psuedo-reaction. This objective function is derived from the turnover of biomass precursors, and formulated in such a way that it provides the proportion that each biomass precursor contributes, in one hour, to a gram of biomass. This results in all flux units, for the reactions in the model, being given a unit of $mmol/gDW/h^{-1}$, where gDW is the gram of dry-weight biomass. A metabolic network may have many objective functions, which can represent different physiological states, biomass compositions or other metabolic phenotypes. To summarise, the final [total] linear optimisation problem for FBA and the analysis of a metabolic network, is an adaptation of Eq. 1.5, defined formally below (Eq. 1.9).

$$\begin{aligned} \text{maximise} \quad & Z = \mathbf{c}^{\intercal} v \\ \text{subject to} \quad & Sv = 0 \\ & lb_i \leq v_i \leq ub_i \end{aligned} \tag{1.9}$$

where indexation i is over all reactions in the metabolic model, Z is the optima for the objective function $\mathbf{c}^T \cdot v$, S is the stoichiometric matrix, and v is the flux distribution vector which is bound by the lower (lb) and upper (ub) flux bounds. A summary of the FBA method, including translation of metabolic model from

biochemical network graph representation to linear optimisation problem, is shown in Fig. 1.7.

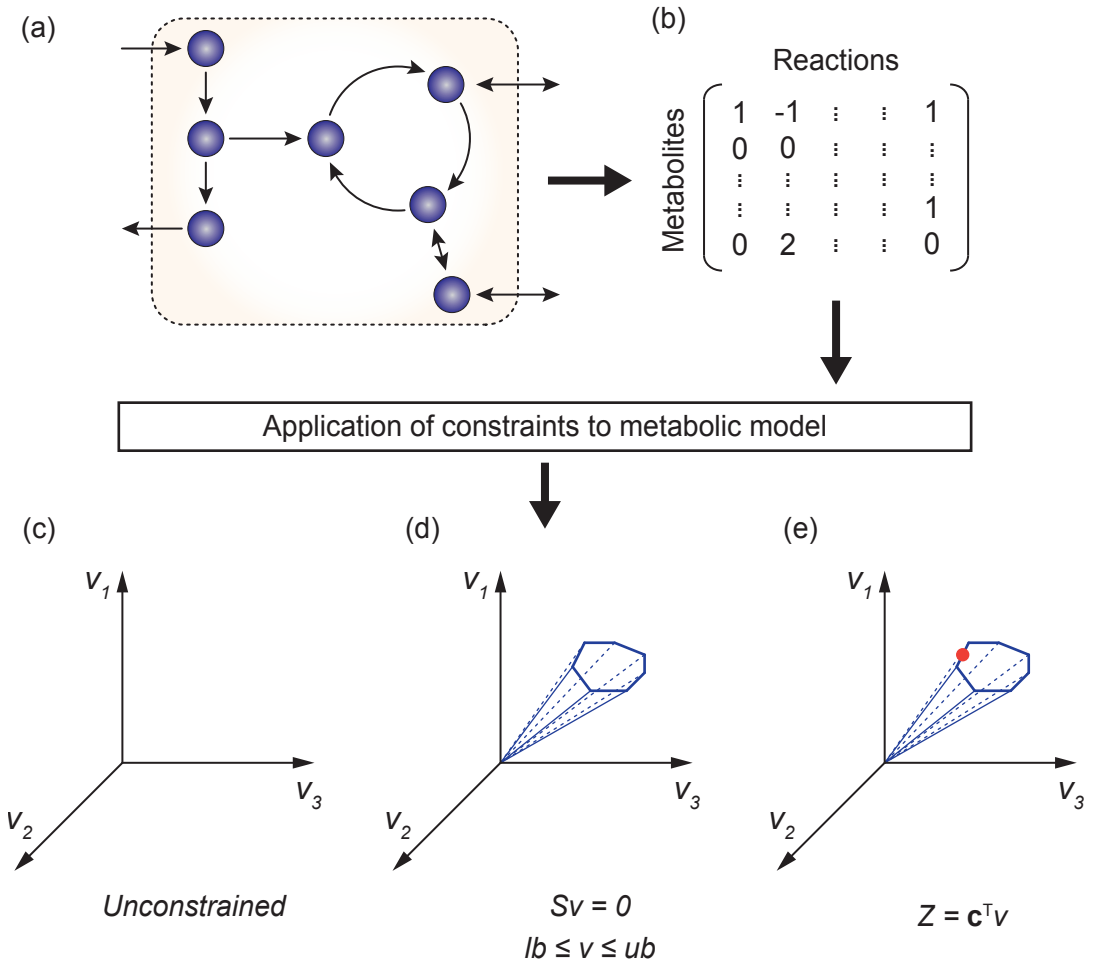


Figure 1.7: Summary of the application of flux balance analysis for metabolic network analysis, from biochemical network to final computational output. (A) a network representation of a small toy metabolic network, with metabolites (circles) and reactions (arrows) shown. (B) a stoichiometric matrix representation of the small toy metabolic network, where values are stoichiometric coefficients of metabolites (rows) partaking in reactions (columns). (C) cartoon representation of the unconstrained linear optimisation problem, with no information regarding reaction minimum and maximum fluxes. (D) cartoon representation of the bounded linear optimisation problem under the constraints of steady-state ($Sv = 0$) and reaction flux (v) lower and upper bounding ($lb \leq v \leq ub$). (E) cartoon representation of the constrained linear optimisation problem with an objective function definition ($Z = \mathbf{c}^{\intercal}v$), yielding an optimal solution (red dot) upon solving the corresponding linear optimisation problem.

FBA is a valuable tool for a variety of scientific industries and academia. It

has been used to predict metabolic fluxes through numerous GSMs for applications in bioengineering (Bonomo & Gill 2005), synthetic biology(Rudge et al. 2012), and manipulation of metabolism for healthcare and pharmaceutical applications(Bordbar et al. 2015, Heinken & Thiele 2015, Shen et al. 2010). However, many advancements and extensions have been created for FBA since its conception, each created for a particular scenario or to increase the accuracy and predictability of the metabolic models.

1.5.3 Expansions of Flux Balance Analysis

FBA has proved to be a useful tool in the analysis of metabolic networks, especially in lieu of data regarding various aspects of the cellular environment that affect the metabolic network (i.e. internal pH, reaction kinetic parameters) (Price et al. 2003, Smallbone & Simeonidis 2009). However, metabolism and by virtue metabolic networks are affected by a number of highly complex and varied parameters, and there are a number of methods developed that expand upon the foundations laid by FBA (Kauffman et al. 2003, Antoniewicz 2015). These mainly focus on the application of additional linear inequalities to the linear optimisation problem, or through integration of other data sets, where available. A brief overview of the various expansions of the FBA methodology, including specific aims of these expansions, is shown in Fig. 1.8.

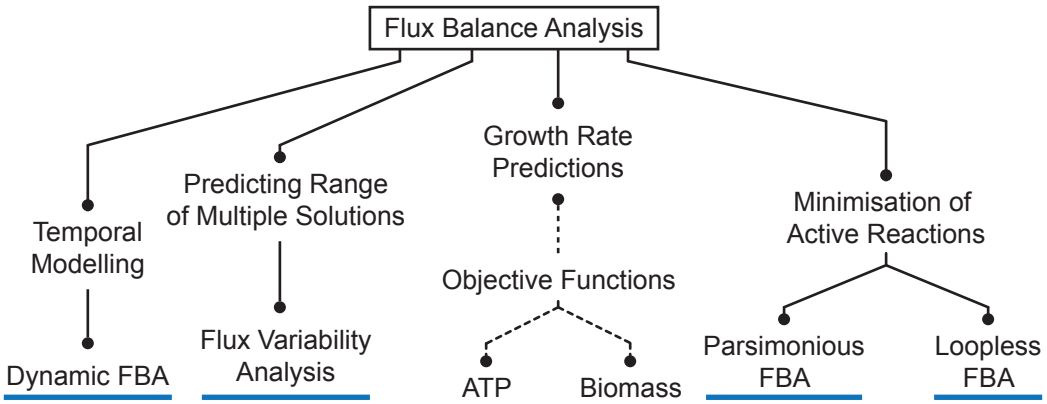


Figure 1.8: Summary of the various expansions of flux balance analysis (FBA). New, expanded FBA methods are highlighted with a blue underline. Solid lines indicate methodology expansions, dashed lines indicate data integration.

1.5.4 Parsimonious FBA

FBA assumes optimality of a metabolic system for a given objective, of which biomass production is often selected as the objective function (Thiele & Palsson 2010, Orth et al. 2010) as it signifies the biomass that is accumulated to facilitate reproduction of an organism, particularly for prokaryotic organisms. This has also been observed to be a driving force in the evolution of bacterial strains (Lenski & Travisano 1994, Ibarra et al. 2002, Molenaar et al. 2009); there is a selection pressure to select for strains that grow the quickest, but also are more efficient in their use of substrates and metabolic network for growth. Combining these ideas yields the assumption that the fastest growing strains have a more efficient enzyme usage.

To achieve this, parsimonious FBA (pFBA) was created as an expansion of FBA. Essentially, pFBA considers the optimal solution as predicted by FBA, whilst attempting to minimise the summation of all gene-associated reaction fluxes, where reactions that are not associated with genes are omitted from such minimisation, and yields the resulting pFBA optima (Lewis et al. 2010). The predicted flux vector (v) yielded by pFBA is considered to be the minimal-flux values that achieve the maximum optima; a set of reactions considered to be [stoichiometric] the most efficient in terms of reaching the desired optima.

As discussed, the basis for the development of pFBA comes from the assumption that biomass accumulation is the driving force of strain selection, and that efficient enzyme use is beneficial for this goal. However, whilst this may apply to prokaryotic organisms, eukaryotic organisms have been and are under different evolutionary pressures (Molenaar et al. 2009). Under this assumption, pFBA appears more applicable to prokaryotic organisms than eukaryotic, which is a consideration that needs to be enacted when designing a metabolic modelling study.

Loopless Flux Balance Analysis

The predictive solutions of FBA (and derivates) are subject to effects of network architecture and reaction parameters. Specifically, thermodynamically infeasible flux distributions can exist, whereby ”loops” of reactions are cycled. To combat this, a version of FBA has been developed that seeks to minimise the number of these loops (Desouki et al. 2015). This loopless FBA (lFBA) is a post-processing step to the constraint-based solution, and removes internal cycles (that is, cycles

that are not associated with the exchange of metabolites) from the flux distribution. As a comparison, for a toy model, of this method and its effect on the flux predictions is shown in Fig. 1.9.

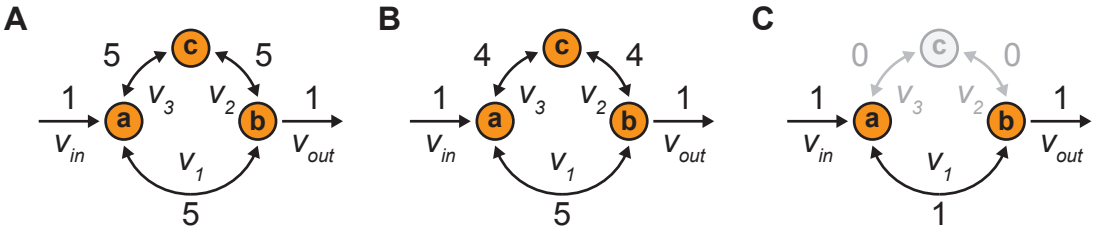


Figure 1.9: Comparison of (**A**) the metabolic network structure, (**B**) the predicted flux distribution with thermodynamically infeasible loops, and (**C**) the predicted flux distribution with elimination of infeasible loops. In this toy model, metabolites a and b can be interconverted either directly or via the c intermediary metabolite. Reaction fluxes (v) are considered reversible, apart from the import (v_{in}) and export (V_{out}) reactions. Metabolites are denoted with lower case letters (a, b, c). Numbers above reaction arrows indicate predicted flux value for the appropriate condition.

There are two main components to lFBA. One is the removal of internal cycles and the preservation of fluxes that are not associated with loops (to preserve optimality of the prediction and network). The second is the minimisation of the sum of absolute fluxes, using the original flux predictions from FBA to bound the fluxes to a feasible range. It should be noted that exchange fluxes are omitted from all steps. The full optimisation problem for lFBA, in conjunction with the initial solution as predicted by FBA (Eq. 1.9), is shown in Eq. 1.10.

$$
\begin{aligned}
\text{min} \quad & \sum_i |v_i| \\
\text{subject to} \quad & \mathbf{c}^\intercal v = Z^0 \\
& Sv = 0 \\
& 0 \leq v_i \leq v_i^0 \qquad \text{for } i : v_i^0 \geq 0 \\
& v_i^0 \leq v_i \leq 0 \qquad \text{for } i : v_i^0 < 0 \\
& v_j = v_j^0 \qquad \text{for exchange fluxes } v_j
\end{aligned}
\tag{1.10}
$$

where indexation i is over all non-exchange or objective reaction fluxes. The original optima value is maintained by setting the value of the objective function to the original (0) FBA solution Z^0. Reaction fluxes are bound so that directionality

of reactions is also conserved; forward reaction fluxes $v_i^0 \geq 0$ and reverse reaction fluxes $v_i^0 < 0$. Exchange fluxes v_j remained unaltered in this process.

In contrast to FVA, which does not enforce the exchange flux conditions or the constraints regarding reaction reversibility, lFBA yields a flux distribution where at least one reaction in every possible cycle is inactive, thus removing the thermodynamically infeasible loops from the prediction. This method has been refined to a level that it can now be ran with little computational cost.

1.5.5 Dynamic FBA

Due to the lack of kinetic parameters and the constraint of the steady-state assumption, FBA cannot be used to model changes in a metabolic system over time (Smallbone et al. 2010, Orth et al. 2010). However, the FBA method has been expanded and adapted to capture some dynamics of a metabolic network by modelling the change in metabolite concentrations over time. This is achieved through the method of dynamic FBA (dFBA).

dFBA requires the relaxation of the steady-state assumption in combination with multiple, individual linear optimisation problems (Eq. 1.9), which represent a set of time intervals. The initial metabolite concentrations, as defined in the original FBA optimisation problem, are used to calculate the subsequent change in metabolite concentration over time. Because it is only the exchange metabolite that have an initial concentration defined, these are the only metabolites that can be simulated as changing in concentration over time. At each optimisation step, the metabolite fluxes are predicted by re-running FBA, and then updating the remaining concentrations of exchange metabolites. These values are then used to update the exchange fluxes, through flux bound alterations, to simulate a time step in a linear fashion. These assumptions, combined with the relaxation of the steady-state assumption, are known as a pseudo steady-state (Mahadevan et al. 2002, Meadows et al. 2010, Hoppe et al. 2007).

Flux Variability Analysis

FBA does not necessarily provide a unique solution for a given objective function (Mahadevan & Schilling 2003), and for any particular optimum there may be many different solutions (i.e. many flux vectors v that satisfy Eq. 1.9). For

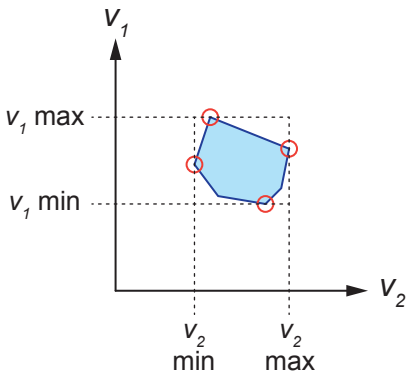


Figure 1.10: Cartoon representation of the flux (v) minimum (min) and maximum (max) values that are calculated for a given optimal solution space (blue shape). The multiple optima (red circles) achieve the same value for the objective function and define the optimal flux range for reactions in the model.

instance, glucose may be converted to biomass precursors through a variety of different pathways or metabolites. Flux variability analysis (FVA) provides an additional layer of information to FBA by predicting the possible flux range for each reaction: the ranges of specific values for a particular reaction flux v_i in which the final optimal solution remains the same (Fig. 1.10).

Under these conditions, the fluxes predicted are the minimum and maximum flux values that a reaction can have, given both the steady-state assumption and the optimality conditions for the system; this is usually the optimal state solution obtained from FBA for a given objective function. more formally, the FVA linear problem can be described as a continuation of the FBA one, laid out in Eq. 1.9. The FVA linear optimisation problem requires two optimisations, a maximisation (max) and a minimisation (min), shown in Eq. 1.11.

$$\begin{aligned} \text{max/min} \quad & v_i \\ \text{subject to} \quad & Sv = 0 \\ & \mathbf{b}^\intercal v \geq \gamma Z^o \\ & lb_i \leq v_i \leq ub_i \end{aligned} \tag{1.11}$$

where $c = b$ (from Eq. 1.9) and $Z^o = \mathbf{b}^\intercal v^0$ is the optimal solution for FBA (Eq. 1.9). γ is a parameter that controls the type of analysis conducted; if $\gamma = 1$ then the optimal state is analysed, whereas if $0 \leq \gamma < 1$ then the sub-optimal state is analysed.

The output of FVA is two flux distribution vectors, one minimum and one maximum flux vector, subsequently predicting the flux range for each individual reaction given the optimality constraints. Generally, these ranges are interpreted in

terms of how the flux range may relate to the optimal solution: wide flux ranges are associated with reactions that are less important to the optimal solution (particularly if the flux range crosses zero), whereas narrow ranges are associated with reactions that are more important to the optimal solution (Motamedian 2015, Mahadevan & Schilling 2003). The reasoning for this trail of thought is that if a reaction has small variance in the flux value it can have, given the current optimality enforced, then small perturbations to this reaction flux will more greatly affect the optima predicted than that of a reaction with large flux variance. Whilst this presents an interesting point of view in terms of how flux values relate to optima predictions, the flux-range to biological mapping is not one that is yet fully explored. The usefulness of FVA lies in the fact that it can predict multiple ranges of solutions for a given optima, which can narrow down exploration (in terms of reaction targets) for further computational exploration (Motamedian 2015, Mahadevan & Schilling 2003).

Computational Toolboxes

Many software packages exist to facilitate the study of stoichiometric metabolic models. The most notable is the Constraint-Based Reconstruction and Analysis (COBRA) toolboxes. As most analyses in this thesis were conducted in a Python-environment, the appropriate COBRA package was used: CobraPy (Ebrahim et al. 2013). A suitable linear optimisation solver was selected for this study, the IBM CPLEX solver. A full set of definitions for both the personal and server computers used to run the simulations are found in Table 1.2.

Table 1.2: Definitions of computational tools and packages used throughout this thesis.

Computer	Description	Version
Personal	Mac OS X	10.13.3
	Python 3	3.5.2
	CobraPy	0.11.3
	CPLEX	12.7.0
Server	Ubuntu	14.04.5 LTS
	Python 3	3.4.3
	CobraPy	0.10.1
	CPLEX	12.7.0

All forms of FBA discussed (FBA, FVA and lFBA) have been utilised throughout this study, with the appropriate method stated for each research component. The majority of the flux predictions were done using lFBA, due to the reduction of thermodynamically infeasible predictions.

1.6 PREVIOUS METABOLIC MODELLING STUDIES

Metabolic modelling, and more specific FBA, have been used in multiple studies across different scientific fields. Most often, FBA has been used as a predictive tool to facilitate bioengineering and metabolic studies, in particular to predict the ability of an organism to produce a particular compound. There have also been a number of explorative studies, with the aim of being able to describe metabolic phenotypes of organisms. Here, an overview of the applications of FBA in previous studies is discussed.

Evolution of Metabolic Systems. Concepts and theories surrounding evolutionary biology, particularly in regards to the emergence of metabolism and metabolic pathways, has been explored numerous times through computational studies (Schmidt et al. 2003). Building upon these studies, there has been a focus on using FBA to explore the adaptability of a metabolic network, most notably for *E. coli*, to be able to identify how horizontal gene transfer may effect the evolution of bacterial metabolic networks (Pál et al. 2005) or the minimal components

required for a functional metabolic network (Pál et al. 2006). The latter presents the most interesting case of metabolism in the context of infection. FBA predictions can be used to establish the minimal set of genes for a given objective, for instance the minimum metabolic genes required for a pathogen to be pathogenic. This, in combination with evolutionary studies, presents a possibility for predictive infection tools using computational modelling through gene analysis of host and pathogen.

Drug Target Predictions. Expanding upon the identification of metabolic genes that are essential to an organisms pathogenicity, further application of metabolic modelling can be used for potential drug target identification. Previous experimental studies have identified that metabolic flux analysis (MFA) can be used in combination with gene expression data to predict the effect of drugs on *Mycobacterium tuberculosis* metabolism(Colijn et al. 2009). Subsequently, this data was combined with predictive metabolic models of *M. tuberculosis* to further predict the effects of mycolic acid on metabolic pathways (such as fatty acid synthesis).

Other pathogenic organisms have also be the subject of metabolic analysis, such as the GSM of *Plasmodium falciparum*. This study demonstrated the viable systems biology cycle of *in silico* drug target predictions (against metabolic genes) and subsequent *in vitro* experimental studies (Plata et al. 2010).

Phage and Virus Infections. Beyond network analysis and drug prediction, metabolic modelling has also been used to understand the interplay between host and pathogen during an infection (Durmuş et al. 2015, Jamshidi & Raghunathan 2015). Specifically, multiple studies have identified that there are significant rearrangements of host metabolic resources in response to infection of *E. coli* with phage (Jain & Srivastava 2009, Birch et al. 2012). In these cases, the phage (MS2 and T7) were described in the metabolic model as a pseudo-reaction that contained the precursors, generated in the same manner as a biomass objective function (Thiele & Palsson 2010), for phage biomass production. This pseudo-reaction becomes the objective function for the phage, and can then be optimised in order to predict how a phage-controlled (or "infected") host metabolic model's flux is re-distributed.

Human Infections. Further to the bacteria-phage (Law et al. 2013) studies discussed above, and other mammalian-pathogen studies (Eisenreich et al. 2013, Ryu et al. 2015, Quek et al. 2010, Aderem et al. 2011), a model was created of a human macrophage infected with the intracellular bacterial pathogen *M. tuberculosis* and analysed using FBA to predict the interactions between the human host and pathogen (Bordbar et al. 2010). A co-optimisation between the human macrophage objective function and the *M. tuberculosis* associated pseudo-reaction was used as the objective function of the model. As bacteria contain their own metabolic networks and machinery, they can be considered in terms of a separate model that is integrated or appended to the existing host metabolic model (Swann et al. 2015). A virus, with no metabolic network or metabolic resources of its own, is entirely dependent on the host metabolic network (Gilbert et al. 2012), hence why virus infection can be modelled using just a pseudo-reaction append into the existing model.

1.7 AIMS AND OBJECTIVES

This thesis aimed to investigate how a virus-infection can be represented in a human genome-scale metabolic model, to aid in antiviral discovery and experimental design (concerning compound selection), and create an antiviral drug systems biology cycle. The rationale for the need of such a system is the lack of antiviral therapeutics for viruses deemed a potential disease or biosecurity risk. This thesis attempts to provide a suitable framework for host-virus metabolic modelling studies and develops strategies that can be utilised by experimentalists to tackle the lack of viable antivirals. This was done by achieving the following objectives:

1. Creation of a method to describe a virus as a metabolic pseudo-reaction that can be used to represent an infected state when integrated into a suitable host metabolic model
2. Construction of an integrated host-virus metabolic model, for species of the Alphavirus and Flavivirus families, to:
   (a) Understand the changes in the metabolic network during an infection
   (b) Identify potential antiviral targets based upon the use of the metabolic network by the virus-controlled system
3. Develop strategies for antiviral discovery through:

(a) Simulation of reaction perturbations via reaction knockouts and other inhibitory methods

(b) Develop an antiviral strategy that considers host viability and attempts to maintain this while disrupting virus production

(c) Evaluate the effect of single and multiple reaction antiviral strategies

4. Provide testable hypothesis from computational simulations, through the identification of novel drug compounds from antiviral modelling predictions

CHAPTER 2

Integrated Host-Virus Metabolic Models of Chikungunya, Dengue and Zika Viruses

2.1 INTRODUCTION

Recent disease-causing emergent viruses, such as Zika virus (Fauci et al. 2005, Mlakar et al. 2016), illustrate the need for rapid characterisation of the interactions between viruses and their potential hosts to facilitate the development and implementation of antiviral strategies. However, this understanding remains a significant challenge due to lack of virology knowledge and scientific resources, such as lack of experimental data; availability and implementation of suitable host-virus systems; and antiviral drug candidate selection. Development of a system that could facilitate across all these fronts would prove valuable in the understanding of these viral infections, but also towards the goal of effective antiviral development (Littler & Oberg 2005, Geisbert & Jahrling 2004).

Viruses are entirely dependent on their hosts' cellular resources for their replication. This relationship is highlighted by observed variations in virus production levels, correlating with cell-to-cell variance in growth rate and phase (Zhu et al. 2009), as well as virus infection leading to changes in host metabolism (Yu et al. 2011*b*). In particular, virus infection leads to significant metabolic alterations in the host, in some cases resulting in up to 3-fold increase in glycolysis rates (El-Bacha et al. 2004, Jain & Srivastava 2009, Yu et al. 2011*a*) and changes in ATP production rates (Zhu et al. 2009). This observation can be seen as an emergent property of the combined host-virus metabolic system and could be related to changes in host cellular demands arising from viral production (Molenaar et al. 2009, Weiße et al. 2015). More specifically, these alterations in host metabolism

upon infection are interpreted either a virus actively manipulating the host system to their advantage (Maynard et al. 2010), or the additional draw of metabolic components for viral production merely resulting in a re-arrangement of host metabolic fluxes.

These re-arrangements alter the metabolic profile of the network, in terms of degradation and synthesis of metabolic compounds; these are known as catabolic and anabolic states, respectively. Catabolism defines the use of metabolic reactions to break-down larger molecules into smaller ones, and is utilised to extract energy from various compounds (i.e. glucose) or for use in anabolic reactions. In contrast, anabolism refers to the synthesis of larger molecules from various smaller ones (i.e. amino acid biosynthesis). Both states of metabolism can be interchangeable and this change can represent a shift in the metabolic network, in terms of resource allocation. For example, in eukaryotic organisms the citric acid cycle is predominantly fed via pyruvate (from the cytosol) although it may be supplemented, or indeed ran entirely independently, by the anabolism of glutamine (DeBerardinis et al. 2007). This type of metabolism, where biopolymers are used to feed metabolic pathways, is known as anaplerosis (in this particular example. this is known as glutamine anaplerosis). Observing the shift in networks, between catabolic and anabolic states, has been highlighted in previous viral infection studies and is something that can also be observed (Thai et al. 2015, 2014, Goodwin et al. 2015), and potentially more well understood, in a metabolic modelling environment.

Previous studies of viruses and their interaction with host metabolism, in the context of metabolic modelling, have focused on the metabolic interactions of *Escherichia coli* bacteria infected with T7 (Birch et al. 2012) and MS2 (Jain & Srivastava 2009) bacteriophage. These studies have demonstrated the potential of a fully realised systems biology approach to host-pathogen interactions, highlighting the low-resource and relatively small computational costs of such a method. To further develop these approaches to a human-virus study, a suitable host metabolic model is selected along with clinically and scientifically important viral pathogens.

A stoichiometric metabolic model of a human macrophage cell was used to establish an integrated host-virus metabolic model for three viruses causing current (or previously) epidemic outbreaks: Chikungunya virus (CHIKV); Dengue virus (DENV); and Zika virus (ZIKV). These are representatives of the virus genera *Alphavirus* (CHIKV) and *Flavivirus* (DENV, ZIKV), which are positive-sense single-strand RNA viruses with simple physical structures (Mukhopadhyay et al. 2005,

Strauss & Strauss 1994). All three of these viruses have been highlighted as being detrimental to global human health and biosecurity (Luz 2016).

Viruses of both families have been observed to infect many different human cell types (Balsitis et al. 2009, Tang 2012), including monocyte-derived macrophage cell lines (Fox et al. 2015, Gollins & Porterfield 1985, Balsitis et al. 2009), and are usually transmitted to humans via arthropod vectors, the most common being mosquitoes of the *Aedes* genus (Garmashova et al. 2007, Lundström 1999).

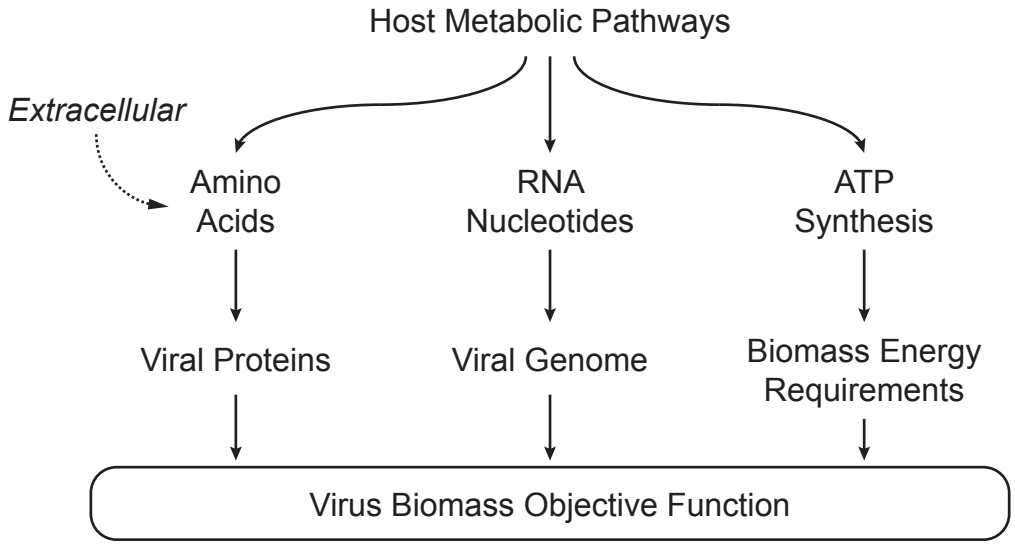


Figure 2.1: Schematic of virus biomass integration with existing host network, and the relevant macromolecules that link them.

The development of host-virus metabolic models, between the human macrophage model (MO) and CHIKV, DENV and ZIKV require the development of a novel methodology to describe viruses as metabolic pseudo-reactions; virus biomass objective functions (VBOFs). The VBOFs capture all the required information about a virus particle to model the infection process within a host, and links the virus production to preexisting metabolic resources allocated by the host (Fig. 2.1). These VBOFs are used to model different virus-infected systems through integration of individual reactions, for each virus, into its own integrated host-virus metabolic model (iHVM). The analysis of the three iHVMs predicts that viral production (in all models) results in alterations to host metabolic fluxes.

2.2 MATERIALS AND METHODS

2.2.1 Virus Species Definition

Chikungunya (*Alphavirus*), Dengue and Zika (*Flavivirus*) viruses (formal species definitions given in Table. 2.1). were selected based upon their disease causing potential and risk to biosecurity (Woolhouse et al. 2016, Lwande et al. 2015, Figueiredo & Figueiredo 2014).

Table 2.1: Virus species used for creation of integrated host-virus metabolic models for characterisation of differences between host and virus optimised metabolic models.

NCBI Reference	Virus Species	Virus Family	NCBI Viral Strain Name	Virus Abbreviation
NC_001449	Chikungunya Virus	*Alphavirus*	S27-African Prototype	CHIKV
NC_001474	Dengue Virus 2	*Flavivirus*	16681	DENV
NC_012532	Zika Virus	*Flavivirus*	MR 766	ZIKV

Genbank files for the viruses were downloaded from the NCBI database (Geer et al. 2010) using the corresponding NCBI reference for each virus species. These files were access on 2017-07-18. These GenBank files contain the nucleotide and amino acid sequences of the genome and virus proteins, respectively.

2.2.2 Generation of virus biomass objective functions

Implementation of the constraint-based modelling approach to study virus infections from a metabolic stance requires the definition of a pseudo reaction that accounts for the production of virus particles from the biomass associated metabolites. These metabolites consist of amino acids and nucleotides, which are further categorised into the proteomic and genomic macromolecular categories, respectively. This pseudo reaction represents the virus biomass objective function (VBOF). To account for metabolic fluxes associated with the virus production, the VBOF captures the stoichiometry of the nucleotides and amino acids required

for virus production. This information is identified from the genome sequence, along with the associated viral protein sequences and how many copies of these proteins are required per viral genome. Additionally, the VBOF also captures the associated energy metabolites (i.e. ATP) required for viral genome and protein production, in a similar manner to biomass production functions used for genome-scale metabolic models (Thiele & Palsson 2010).

Defining Metabolite Stoichiometries

The metabolite stoichiometries required for the VBOF are derived from the viral genome sequence, the subsequently encoded proteins, the copy number of those proteins, and knowledge of the energetic requirements for the formation of peptide bonds (amino acid polymerisation) and phosphodiester bonds (RNA nucleotide polymerisation). For this modelling approach, we do not include in the VBOF a metabolic component that describes the virus envelopes or virus-associated lipids. This is due to a lack of information regarding the stoichiometry of virus associated lipids for species of the *Alphavirus* and *Flavivirus* families.

Genomic and Proteomic Information for the Viruses

The approach outlined in this Methods is best suited to viruses with known nucleotide and amino acid stoichiometry and simple, symmetrical capsid structures. *Alphavirus* and *Flavivirus* virus families meet these criteria and as such mean that Chikungunya, Dengue and Zika viruses are suitable for this modelling approach.

Viruses can be classified by their replication methods, known as the Baltimore Classification System (Yu et al. 2013). Depending upon the classification of a virus, the viral particle may contain more than a single copy of the virus genome. As such, this is factored into the calculation of the nucleotide counts for a given VBOF. All viruses presented in this study, unless explicitly stated otherwise, fall into the Group IV classification: the positive single-stranded RNA (+ssRNA) genome is replicated via a negative ssRNA (-ssRNA) intermediate. Therefore, the overall count of nucleotides required to produce a virus particle are the combined counts of the nucleotides in the +ssRNA sequence and the complementary nucleotides in the -ssRNA sequence, as per the relationships defined in Table 2.2.

Table 2.2: Relationship of nucleotide counts between positive and negative single-sense RNA virus genomes.

Positive RNA Genome Nucleotide	Complementary Negative RNA Genome Nucleotide
Adenosine	Uracil
Cytidine	Guanine
Guanine	Cytidine
Uracil	Adenosine

The count for each RNA nucleotide (adenosine (A), cytidine (C), guanine (G) and uracil (U)) can be taken directly from the virus genome sequence: RNA utilises U in place of thymine (T) in DNA sequences; therefore T is replaced with U for virus genome sequence readouts.

All viruses in this study, unless explicitly stated otherwise, are comprised of two categories of protein that compose the virus proteome: structural and non-structural. The amino acid sequence of these, and indeed any genome derived protein sequence, are obtained from gene annotations of the viral genomes as indicated in the NCBI genome entries (defined in each subsequent chapter). The number of individual proteins, per virus particle, within these categories, is known as the copy number. Copy numbers differ between virus species/ genera/families. For viruses used in this study (from the *Alphavirus* and *Flavivirus* virus families), the structural and nonstructural proteins are expressed in a ratio of copy numbers that are derived from the overall virus structure (i.e. proteins in the capsid and the nucleocapsid). Copy numbers for both of these virus families, for both protein categories and the virus genome are listed in Table 2.3 (Strauss & Strauss 1994, Mukhopadhyay et al. 2005). More broadly, the ratio of different protein classes in a single virus particle can be derived from the overall virus structure or directly from literature and experimental evidence.

Table 2.3: Copy numbers per virus particle for *Alphavirus* and *Flavivirus* virus families.

Virus Component	Alphavirus	Flavivirus
Genome	1	1
Nonstructural Protein	1	1
Structural Protein	240	180

Once the genomic and proteomic composition of a virus particle has been identified, a VBOF can be created. This is done through the assignment of stoichiometric coefficients for each biomass associated metabolite (amino acids, nucleotides and the energy requirement metabolites). These stoichiometric coefficients are defined using the per virus particle frequency of the associated metabolite (i.e. adenosine count for nucleotides). The full method for calculating all virus macromolecular components are detailed below.

Calculating nucleotide investment per virus

The total moles of each nucleotide in a mole of virus particle (N_i^{TOT}) are obtained from their frequency in the virus genome (N_i^G) and replication intermediates (N_i^R), multiplied by the genome copy number (C_g):

$$N_i^{TOT} = C_g \left(N_i^G + N_i^R \right) \tag{2.1}$$

where the indexation i is over nucleotides. The moles of nucleotides are then converted into grams of nucleotides per mole of virus ($g_{NTPS} mol_{virus}^{-1}$; G_i^N) by multiplying (N_i^{TOT}) with the respective molar mass (g mol^{-1}) of the nucleotides (M_i^N):

$$G_i^N = N_i^{TOT} M_i^N \tag{2.2}$$

where the indexation i is over nucleotides. Summing G_i^N over all nucleotides and combining this with the similar calculation for amino acids calculates the total molar weight of the virus in terms of the main macromolecular constituents (M_v, see Equation 2.15). Finally, the stoichiometric coefficients of each nucleotide in the

VBOF are expressed as millimoles per gram of virus ($mmol_{NTPS}g_{virus}^{-1}$; S_i^N):

$$S_i^N = 1000\left(\frac{N_i^{TOT}}{M_v}\right) \tag{2.3}$$

where the indexation i is over nucleotides.

Calculating amino acid investment per virus

The total moles of each amino acid per mole of virus particle (X_j^{TOT}) is obtained using the sequence information of the structural (X_j^{SP}) and non-structural (X_j^{NP}) proteins. Frequencies of each amino acid in these proteins are multiplied by the respective copy numbers (C_{SP} and C_{NP} respectively):

$$X_j^{TOT} = C_{SP}\left(X_j^{SP}\right) + C_{NP}\left(X_i^{NP}\right) \tag{2.4}$$

where the indexation j is over amino acids. Copy numbers are as previously defined in Table 2.3. The moles of amino acid per mole of virus are then converted into grams of amino acid per mole of virus ($g_{AA}mol_{virus}^{-1}$; G_j^X) by multiplying (X_j^{TOT}) with the respective molar mass (g mol^{-1}) of each amino acid (M^X):

$$G_j^X = X_j^{TOT} M_j^X \tag{2.5}$$

where the indexation j is over amino acids. Finally, the stoichiometries of each amino acid in the VBOF are expressed as millimoles per gram of virus ($mmol_{AA}g_{virus}^{-1}$; S_j^X):

$$S_j^X = 1000\left(\frac{X_j^{TOT}}{M_v}\right) \tag{2.6}$$

where the indexation j is over amino acids.

Calculating the energy requirements per virus

Calculating ATP requirement for amino acid polymerisation ($mmol\ g_{virus}^{-1}$). The polymerisation of amino acid monomers requires approximately 4 ATP molecules per peptide bond (Haynie 2009), defined as the constant k_{ATP} (= 4). The overall moles of ATP (A^{TOT}) required to form the structural (A^{SP}) and non-structural

(A^{NP}) proteins are calculated from the respective amino acid counts:

$$A^{SP} = \left(\sum_j X_j^{SP} \cdot k_{ATP} \right) - k_{ATP} \tag{2.7}$$

$$A^{NP} = \left(\sum_j X_j^{NP} \cdot k_{ATP} \right) - k_{ATP} \tag{2.8}$$

$$A^{TOT} = C_{SP} \left(A^{SP} \right) + C_{NP} \left(A^{NP} \right) \tag{2.9}$$

where the indexation is over amino acids. From A^{TOT}, the stoichiometry of ATP in the VBOF as millimoles per gram of virus (S^{ATP}):

$$S^{ATP} = 1000 \left(\frac{A^{TOT}}{M_v} \right) \tag{2.10}$$

The polymerisation of the amino acids forms peptide bonds, which requires the hydrolysis of ATP. The water requirement for Polymerisation (S^{H_2O}) is equal to that of ATP (S^{ATP}). The products from the hydrolysis of ATP (ADP, P_i and H^+) are also accounted for in the VBOF with negative stoichiometric terms that are equal to that of ATP (see Equation 2.16).

Calculating the pyrophosphate (PP_i) liberation from RNA nucleotide polymerisation ($mmol\ g_{virus}^{-1}$) . The polymerisation of RNA nucleotide monomers to form the viral genome (+ssRNA or -ssRNA) liberates a PP_i molecule (Haynie 2009), defined as the constant k_{PPi} (=1). The overall moles of PP_i (P^{TOT}) required to form the viral genome (P^G) and replication intermediates (P^R) are calculated from the respective nucleotide counts:

$$P^G = \left(\sum_i N_i^G \cdot k_{PPi} \right) - k_{PPi} \tag{2.11}$$

$$P^R = \left(\sum_i N_i^R \cdot k_{PPi} \right) - k_{PPi} \tag{2.12}$$

$$P^{TOT} = C_g \left(P^G + P^R \right) \tag{2.13}$$

where the indexation i is over nucleotides. Conversion of this value into the PP_i stoichiometry for the VBOF as millimoles per gram of virus (S^{PPi}) requires the use of the overall molar mass (g mol^{-1}) of one mole of virus:

$$S^{PPi} = 1000 \left(\frac{P^{TOT}}{M_v} \right) \tag{2.14}$$

Calculating total viral molar mass

The total molar mass of the virus (M_v) is calculated from the total mass of the genomic and proteomic components:

$$M_v = \left(\sum_i G_i^N \right) + \left(\sum_j G_j^X \right) \tag{2.15}$$

where i, j are indexation over nucleotides and amino acids (respectively).

Final construction of the virus biomass objective function

Left and right terms of the VBOF reaction are based on the above calculations of the nucleotide, amino acid and energy requirement metabolite stoichiometric coefficients. The final stoichiometry for the VBOF pseudo-reaction is:

$$S_i^N + \ldots + S_j^X + \ldots + S^{ATP} + S^{H_2O} \rightarrow S^{ADP} + S^{Pi} + S^{H^+} + S^{PPi} \tag{2.16}$$

This pseudo-reaction accounts for the virus' biomass and the energy requirements associated with its production. The VBOF is incorporated into stoichiometric metabolic models representative of a host organism to simulate the metabolic demands of a virus within that system. The full VBOF workflow is summarised in Fig. 2.2.

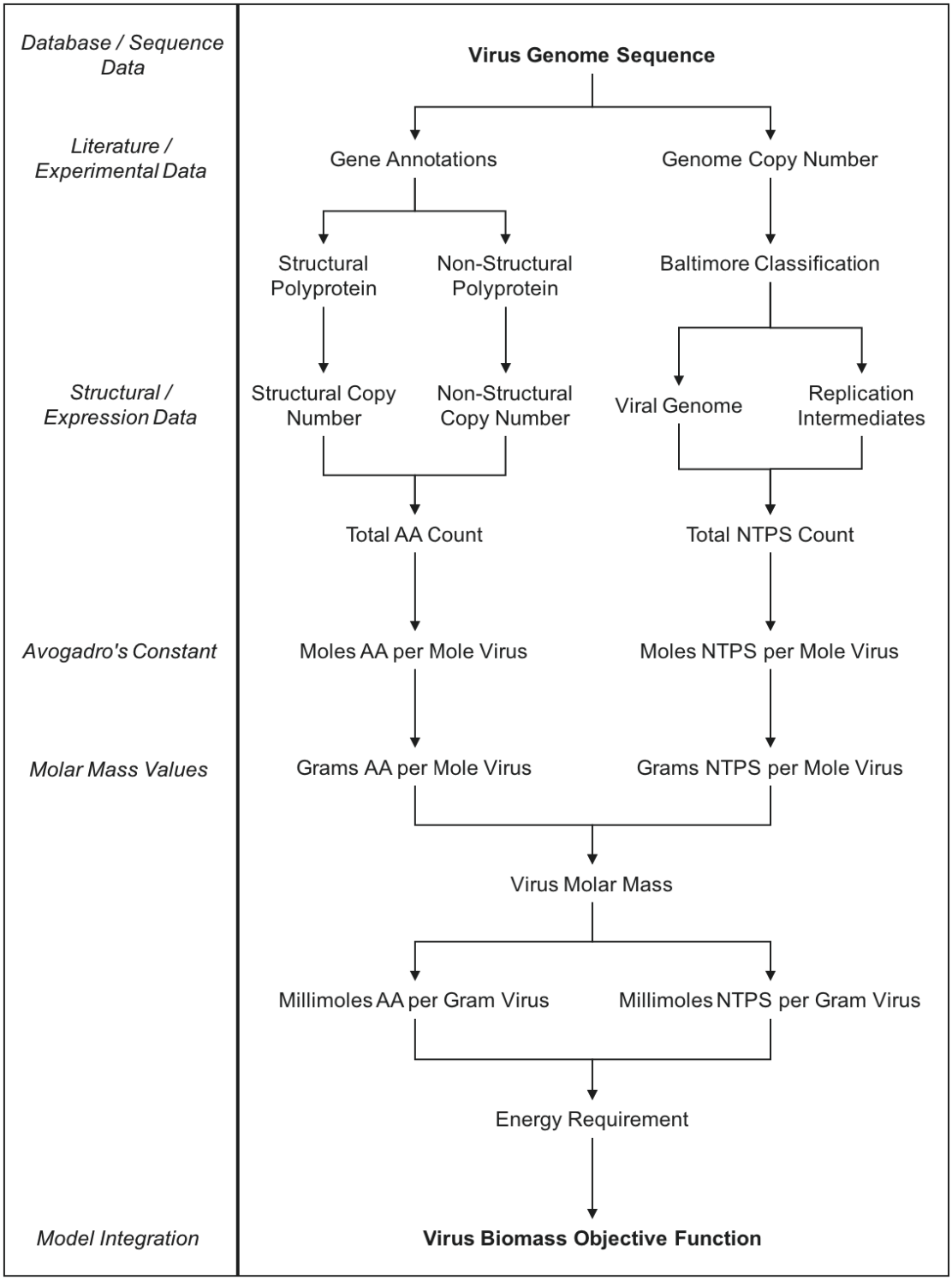


Figure 2.2: Schematic of virus biomass objective function (VBOF) generation. Diagram outlines the process of forming the necessary components for the pseudo-reaction that represents the production of virus particles (biomass).

2.2.3 Human Macrophage Metabolic Model

To model the host-virus metabolic interactions of Chikungunya, Dengue and Zika virus infections, a suitable model was selected. Experimental evidence has shown that human monocyte-derived macrophage cell lines can support the growth of viruses from *Alphavirus* and *Flavivirus* families (Gollins & Porterfield 1985, Fox et al. 2015, Balsitis et al. 2009). The human macrophage model iAB-AMO-1410 (Bordbar et al. 2010) was selected as the suitable host metabolic model for this study.

Model Definition

The reconstruction of the human macrophage metabolic model iAB-AMO-1410 is detailed in a previous publication (Bordbar et al. 2010). Briefly, this reconstruction is generated using gene expression data from human alveolar macrophages ($n = 11$) obtained from the Gene Expression Omnibus (Kazeros et al. 2008). This expression data is used to derive the human macrophage cell type-specific metabolic model from the global human metabolic network reconstruction, Recon 1 (Thiele et al. 2013). The breakdown of the iAB-AMO-1410 model components are listed in Table 2.4.

Table 2.4: Components of the human macrophage metabolic model iAB-AMO-1410.

Model Description	Number
Genes	1797
Reactions	3394
Subsystems	96
Metabolites	2572
Compartments	8

In the context of this model, only metabolic genes are characterised. These genes are associated with a reaction, which is defined as the interactions between metabolites within the metabolic network. These reactions can be grouped into subsystems, which are defined as a group of reactions that are attributed to the same metabolic pathway per the KEGG notation (Kanehisa et al. 2017). Subsystems

of the iAB-AMO-1410 metabolic model, along with some associated reactions, are detailed in Table 2.5.

Table 2.5: Subsystems of the human macrophage metabolic model iAB-AMO-1410 (ranked in alphabetical order).

Model Subsystem	Number of Reactions
Alanine and Aspartate Metabolism	8
Alkaloid biosynthesis II	3
Aminosugar Metabolism	19
Arginine and Proline Metabolism	21
Ascorbate and Aldarate Metabolism	13
Beta-Alanine metabolism	8
Bile Acid Biosynthesis	28
Biotin Metabolism	6
Blood Group Biosynthesis	39
Butanoate Metabolism	3
C5-Branched dibasic acid metabolism	4
Carnitine shuttle	109
Cholesterol Metabolism	42
Chondroitin / heparan sulfate biosynthesis	44
Chondroitin sulfate degradation	26
Citric Acid Cycle	17
CoA Biosynthesis	6
CoA Catabolism	4
CYP Metabolism	6
Cysteine Metabolism	15
D-alanine metabolism	3
Demand Reaction	29
Eicosanoid Metabolism	21
Fatty acid activation	36
Fatty acid elongation	21
Fatty Acid Metabolism	26
Fatty acid oxidation	29
Fatty acid oxidation, peroxisome	19
Folate Metabolism	50
Fructose and Mannose Metabolism	16

Continued on next page

Table 2.5 – continued from previous page

Model Subsystem	Number of Reactions
Galactose metabolism	10
Glutamate metabolism	11
Glutathione Metabolism	11
Glycerophospholipid Metabolism	48
Glycine, Serine, and Threonine Metabolism	22
Glycolysis/Gluconeogenesis	28
Glycosylphosphatidylinositol (GPI)-anchor biosynthesis	31
Glyoxylate and Dicarboxylate Metabolism	9
Heme Biosynthesis	5
Heme Degradation	2
Heparan sulfate degradation	27
Histidine Metabolism	11
Hyaluronan Metabolism	5
IMP Biosynthesis	10
Inositol Phosphate Metabolism	59
Keratan sulfate biosynthesis	59
Keratan sulfate degradation	60
Limonene and pinene degradation	6
Lysine Metabolism	14
Media Exchange	381
Methionine Metabolism	5
Miscellaneous	10
N-Glycan Biosynthesis	108
N-Glycan Degradation	15
NAD Metabolism	23
None	3
Nucleic acid degradation	9
Nucleotide Sugar Metabolism	2
Nucleotides	163
O-Glycan Biosynthesis	7
Others	2
Oxidative Phosphorylation	7
Pentose and Glucuronate Interconversions	11

Continued on next page

Table 2.5 – continued from previous page

Model Subsystem	Number of Reactions
Pentose Phosphate Pathway	16
Phenylalanine metabolism	4
Propanoate Metabolism	9
Purine Catabolism	9
Pyrimidine Biosynthesis	8
Pyrimidine Catabolism	14
Pyruvate Metabolism	30
R Group Synthesis	42
Riboflavin Metabolism	3
ROS Detoxification	6
Salvage Pathway	3
Selenoamino acid metabolism	11
Sphingolipid Metabolism	62
Starch and Sucrose Metabolism	20
Steroid Metabolism	40
Taurine and hypotaurine metabolism	3
Tetrahydrobiopterin	9
Thiamine Metabolism	6
Transport, Endoplasmic Reticular	93
Transport, Extracellular	505
Transport, Golgi Apparatus	70
Transport, Lysosomal	68
Transport, Mitochondrial	181
Transport, Nuclear	60
Transport, Peroxisomal	73
Triacylglycerol Synthesis	13
Tryptophan metabolism	37
Tyr, Phe, Trp Biosynthesis	1
Tyrosine metabolism	31
Ubiquinone Biosynthesis	14
Urea cycle/amino group metabolism	22
Valine, Leucine, and Isoleucine Metabolism	30
Vitamin A Metabolism	29

Continued on next page

Table 2.5 – continued from previous page

Model Subsystem	Number of Reactions
Vitamin B12 Metabolism	2
Vitamin B6 Metabolism	9
Vitamin D	14

The stoichiometric matrix is derived from the reactions within the model, and are written such that metabolites with negative stoichiometric coefficients are left-side entities and metabolites with a positive stoichiometric coefficient are considered right-hand entities. Reactions may be reversible (they can proceed in either forward or reverse direction, and therefore have a positive or negative flux value respectively) and this reversibility is incorporated via the reaction bounds (Table 2.6).

Table 2.6: Definition of reaction reversibility in the metabolic model.

Reaction Description	Reaction Bound Conditions
Non-reversible	lower bound = 0; upper bound > 0
Non-reversible	lower bound < 0; upper bound = 0
Reversible	lower bound < 0; upper bound > 0

The metabolic model consists of several compartments; distinct areas of the network that represent the various compartments, which represent different organelles, of the macrophage cell. These compartments are abbreviated using a single letter and are appended to each model metabolite identifier to indicate compartment-specific metabolites (Table 2.7. While metabolites can have identical identifiers, the compartment abbreviation indicates (regarding model entities) that these are not the same (i.e. 'ATP(c)' and 'ATP(m)' represent the cytosolic and mitochondrial versions of ATP and are considered as separate metabolic entities).

Table 2.7: Compartment definitions for the iAB-AMO-1410 metabolic model.

Model Metabolite Compartment	Abbreviation
Cystosol	*(c)*
Extracellular	*(e)*
Golgi apparatus	*(g)*
Lysosome	*(l)*
Mitochondria	*(m)*
Nucleus	*(n)*
Endoplasmic reticulum	*(r)*
Peroxisome	*(x)*

Reactions and metabolites can be interchanged between compartments, in what is defined as a transport reaction. A select subset of these transport reactions are the exchange reactions. These are responsible for the defining the availability of metabolites in the extracellular *in silico* media for uptake by the metabolic model.

Media Definition

Flux balance analysis of stoichiometric metabolic models requires the definition of an *in silico* media. This is defined using a set of exchange reactions that import metabolites into the modelling environment to be accessible to the metabolic model. This import is defined (as with any other reaction in this modelling approach) as a flux rate that has a non-zero lower bound. Contrary to most reactions in the metabolic model, the exchange reactions are written with the imported metabolite ($m\,(e)$) as a left-hand term, and the right-hand term is empty set (Eq. 2.17).

$$m\,(e) \rightarrow \varnothing \tag{2.17}$$

Using this reaction notation, import of the exchange metabolite is denoted as a negative flux value. The model consists of 382 exchange reactions, of which 27 have a non-zero lower bound. This indicates that these reactions form the *in silico* importable media (Table 2.8). These exchange reaction constraints are derived from

previous literature that explored *in vitro* murine macrophage metabolite media uptake rates (Newsholme et al. 1999, 1986, Sato et al. 1987).

Table 2.8: *In silico* Media Definitions for iAB-AMO-1410.

Exchange Reaction ID	Exchange Metabolite Name	Exchange Reaction Lower Bound
EX_hco3(e)	Bicarbonate	-10
EX_but(e)	Butyrate (n-C4:0)	-0.0038
EX_ca2(e)	Calcium	-inf
EX_cl(e)	Chloride	-inf
EX_glc(e)	D-Glucose	-0.2718
EX_h(e)	Hydrogen	-inf
EX_arg-L(e)	L-Arginine	-0.02375
EX_gln-L(e)	L-Glutamine	-0.0765
EX_his-L(e)	L-Histidine	-0.01
EX_ile-L(e)	L-Isoleucine	-0.01
EX_leu-L(e)	L-Leucine	-0.0362216
EX_lys-L(e)	L-Lysine	-0.01
EX_met-L(e)	L-Methionine	-0.01
EX_phe-L(e)	L-Phenylalanine	-0.01
EX_thr-L(e)	L-Threonine	-0.01
EX_trp-L(e)	L-Tryptophan	-0.01
EX_val-L(e)	L-Valine	-0.01
EX_ocdca(e)	Octadecanoate (n-C18:0)	-0.1
EX_ocdcea(e)	Octadecenoate (n-C18:1)	-0.0192
EX_o2(e)	Oxygen	-0.3066
EX_pi(e)	Phosphate	-10
EX_k(e)	Potassium	-inf
EX_pyr(e)	Pyruvate	-0.0568
EX_na1(e)	Sodium	-inf
EX_so4(e)	Sulfate	-inf
EX_ttdca(e)	Tetradecanoate (n-C14:0)	-0.1
EX_h2o(e)	Water	-inf

Most metabolites exchanged between the *in silico* media, and the model are bounded with non-zero, non-infinite fluxes. These metabolites do not have a limitless supply and are considered constrained. Metabolites such as hydrogen, water and some ions are considered to be in abundant supply; this is denoted as $-inf$ in the exchange reaction lower bounds. All exchange reactions also have an upper bound value of $+inf$, indicating that all these metabolites are exportable from the model. It should be noted that these "export reactions" are considered an

abstraction: the exchange reactions are artificial reactions, in that they do not arise from the organisms genome and associated metabolic genes, that define the availability of metabolites in the accessible extracellular environment.

Objective Function Definition

The objective function for iAB-AMO-1410 was formulated using the standard biomass objective function methods for metabolic model generation (Bordbar et al. 2010, Feist & Palsson 2010, Thiele & Palsson 2010). The objective function considers the maintenance of cell biomass, as opposed to the more commonly used accumulation of biomass(Feist & Palsson 2010). The latter is often assumed for microbial metabolic models; in mammalian models, the maintenance of biomass for cell viability is selected as a more appropriate objective.

The iAB-AMO-1410 objective function, named 'biomass_mac' in the metabolic model, consists of several metabolite categories, known as macromolecule types, and considered concerning total biomass objective function abundance (Fig. 2.3). The abundance of the different macromolecule types, concerning a whole macrophage cell, were obtained from literature: protein and sugars (Iyengar & UK 1985), lipids (Schmien et al. n.d.), and DNA (Grutman & Orgel 1970). While RNA is also present in the macrophage biomass, during model creation there was no literature on RNA content; therefore the overall RNA content was adapted from yeast and mouse hybridoma reconstructions (Sheikh et al. 2005).

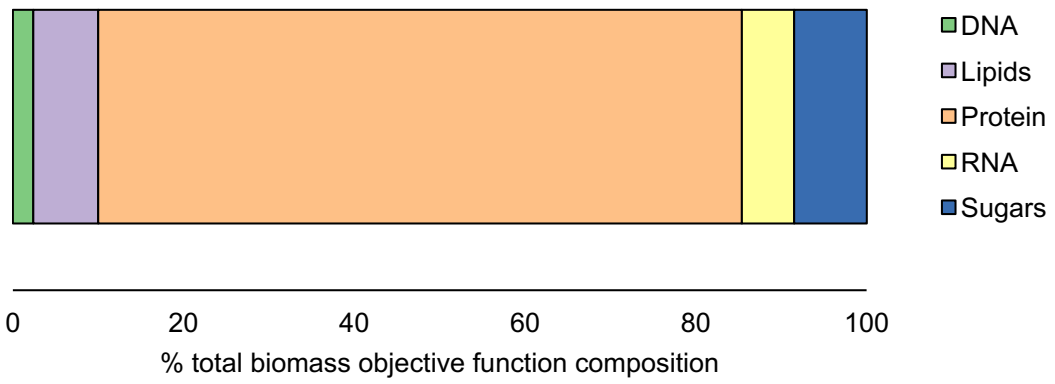


Figure 2.3: Breakdown of the human macrophage (iAB-AMO-1410) biomass objective function for macrophage biomass maintenance in terms of macromolecular types. Full breakdown of each individual type is listed in Appendix B.

2.2.4 Additional Pseudo-reactions

In addition to the objective function that is added to the model, other artificial (reactions that are not identified from the organisms annotated genome) are also present within the metabolic model. These reactions are mainly comprised of the exchange reactions that supply the *in silico* media. They also consist of cellular transport reactions (i.e. between intracellular compartments or between the cellular system and external media), spontaneous reactions and disassociations.

Integration of model with virus biomass objective functions

The individual integrated host-virus stoichiometric metabolic models were created for CHIKV, DENV and ZIKV (MO-CHIKV, MO-DENV and MO-ZIKV respectively) through the integration of the respective VBOFs into a virus-specific instance of the iAB-AMO-1410 human macrophage metabolic model. The VBOF is appended to the iAB-AMO-1410 metabolic model stoichiometric matrix so that the matrix S is redefined as m x $n + 1$.

2.2.5 Metabolic Modelling Computational Parameters

Flux Balance Analysis

All simulations were performed within the Python version 3.5.2 environment (Millman & Aivazis 2011). FBA and all associated FBA-variants were run using the Python package cobraPy version 0.10.1 (Ebrahim et al. 2013) with IBM CPLEX optimisation studio version 12.7.0 as the linear optimisation solver (Barsoum et al. 2008). The FBA performed in this chapter was the loopless FBA variant (Desouki et al. 2015). Fluxes predicted from FBA of the metabolic models are reported either in the original units of mmol.gDw.h-1 or normalised to the glucose exchange rate. The glucose exchange rate is defined as the predicted flux absolute value for the glucose exchange reaction EX_glc(e), and all flux values are normalised to the absolute value of it. The redefined unit, in this case, is mmol/glucose/h-1.

Host and Virus Optimised Network Definitions

For the modelling approach, the model is only considered as either host-optimal or virus-optimal. The latter represents a thought experiment where the virus has complete control of the metabolic rearrangements of the host's network, without maintaining host function, for viral production.

Host optimised network is defined as the host-virus integrated metabolic model optimised for the pseudo-reaction objective function 'biomass_mac'. The corresponding optimisation results in the host-optimal network state and associated predicted flux values for the reactions in this model.

Virus optimised network is defined as the host-virus integrated metabolic model optimised for the VBOF. The corresponding optimisation results in the virus-optimal network state and associated predicted flux values for the reactions in the model.

2.2.6 Host-Virus Metabolic Comparisons

Comparisons between host and virus usage of the integrated host-virus metabolic network can be performed with analysis which includes a combination of their (respective) biomass pseudo-reaction objective functions and the optimisation of the metabolic network for host and virus.

Characterisation of the stoichiometric differences between host and virus

Comparisons between the metabolite stoichiometries in the host and virus pseudo-reactions quantify the differences in amino acid and nucleotide requirements that fulfil the host or virus objective functions. The individual stoichiometric coefficients for all metabolites that are not associated with energy requirements (section 2.2.2) are normalised against the sum of stoichiometries of metabolites present in the objective function (Eq. 2.18). In the case of ATP only the nucleotide associated stoichiometric coefficient is considered, the calculated energy requirement ATP stoichiometric value is not evaluated.

$$\widetilde{S}_i = \frac{S_i^V}{\sum_k S_k^V} \tag{2.18}$$

where the indexation i is over nucleotides or amino acids, and k is over all biomass precursors (as defined above). These normalised stoichiometric values ($\widetilde{S}$) are then used to calculate the $\log_2$ fold-change (FC) of amino acids and nucleotides between host and virus pseudo-reactions (Eq. 2.19).

$$log_2\left(\frac{\widetilde{S}_i^V}{\widetilde{S}_i^H}\right) \tag{2.19}$$

where indexation i is over nucleotides or amino acids, H and V indicate the use of host and virus biomass function associated normalised stoichiometries, respectively. A positive value indicates a higher usage of metabolite i by the virus than the host, while a negative value indicates a lower usage.

Comparison of host and virus optimised states

The integrated host-virus metabolic models are optimised for the host and virus objective functions resulting in predictions of sets reaction flux vectors that satisfy the host-optimal and virus-optimal networks, respectively. This is done using the loopless FBA approach, as it has been shown to be more robust to instabilities associated with prediction and comparison of single optimal flux sets (Desouki et al. 2015). Comparisons between host and virus optimised systems utilise the predicted flux values (f) for host and virus optimised metabolic networks.

Glucose Normalised Flux Results for individual reactions are compared by normalising the reaction flux to the predicted optimal glucose uptake flux (f^g). This is done in order to provide a sensible comparison between the host- and virus-optimised states, but also to put the predicted fluxes in context of glucose usage. This is defined as the flux value for the 'EX_glc(e)' metabolic model reaction. The normalisation of reaction fluxes is defined in Eq. 2.20

$$\widetilde{f}_i = \frac{f_i}{|f^g|} \tag{2.20}$$

where the indexation i is over all reactions. Superscript g denotes the flux associated with the glucose exchange reaction. Glucose normalised flux values $\widetilde{f}$ are then reported for each reaction. This is performed for both host- and virus-optimised metabolic networks.

2.2.7 Development of ViraNet package

To facilitate the creation, analysis and processing of these integrated host-virus metabolite models (iHVMs) an in-house Python package was developed to automate the procedure of VBOF creation, model integration and comparison of host and virus optimised states. This package is titled ViraNet, and is built utilising existing Python packages: cobraPy for metabolic modelling approaches (including FBA and variants) (Ebrahim et al. 2013), NumPy for mathematical operations (van der Walt et al. 2011), and Pandas for dataset processing (McKinney 2010). This package can automatically construct a VBOF from the GenBank file for viruses of the *Alphavirus* and *Flavivirus* families, and integrate with a user-supplied stoichiometric model. The package then performs the analyses outlined in this chapter. The code for ViraNet is supplied in Appendix A.5.

2.2.8 Creation of virus biomass objective functions and integrated host-virus metabolic models

To analyse virus infections regarding the metabolic interactions between the viruses CHIKV, DENV and ZIKV and the host human macrophage iAB-AMO-1410, a set of VBOFs were created and integrated into three separate stoichiometric metabolic models. A full description of each of the VBOF pseudo-reactions, including biomass metabolite stoichiometric coefficients, are detailed below.

Chikungunya Virus Biomass Objective Function Equation (CHIKV_bm)

$$0.49\,\mathrm{ala_{(c)}} + 0.27\,\mathrm{arg_{(c)}} + 0.25\,\mathrm{asn_{(c)}} + 0.21\,\mathrm{asp_{(c)}} + 23.94\,\mathrm{atp_c} + 0.11\,\mathrm{ctp_c} + 0.23\,\mathrm{cys_{(c)}} + 0.25\,\mathrm{gln_{(c)}} + 0.29\,\mathrm{glu_{(c)}} + 0.36\,\mathrm{gly_{(c)}} + 0.11\,\mathrm{gtp_{(c)}} + 23.82\,\mathrm{h_2o_{(c)}} + 0.20\,\mathrm{his_{(c)}} + 0.29\,\mathrm{ile_{(c)}} + 0.39\,\mathrm{leu_{(c)}} + 0.37\,\mathrm{lys_{(c)}} + 0.14\,\mathrm{met_{(c)}} + 0.16\,\mathrm{phe_{(c)}} + 0.43\,\mathrm{pro_{(c)}} + 0.34\,\mathrm{ser_{(c)}} + 0.43\,\mathrm{thr_{(c)}} + 0.07\,\mathrm{trp_{(c)}} + 0.23\,\mathrm{tyr_{(c)}} + 0.11\,\mathrm{utp_{(c)}} + 0.48\,\mathrm{val_{(c)}} \longrightarrow 23.82\,\mathrm{adp_{(c)}} + 23.82\,\mathrm{h_{(c)}} + 23.82\,\mathrm{pi_{(c)}} + 0.46\,\mathrm{ppi_{(c)}}$$

Dengue Virus Biomass Objective Function Equation (DENV_bm)

$$0.23\,\mathrm{ala_{(c)}} + 0.26\,\mathrm{arg_{(c)}} + 0.18\,\mathrm{asn_{(c)}} + 0.13\,\mathrm{asp_{(c)}} + 19.67\,\mathrm{atp_{(c)}} + 0.16\,\mathrm{ctp_{(c)}} + 0.11\,\mathrm{cys_{(c)}} + 0.16\,\mathrm{gln_{(c)}} + 0.30\,\mathrm{glu_{(c)}} + 0.44\,\mathrm{gly_{(c)}} + 0.16\,\mathrm{gtp_{(c)}} + 19.48\,\mathrm{h_2o_{(c)}} + 0.11\,\mathrm{his_{(c)}} + 0.32\,\mathrm{ile_{(c)}} + 0.43\,\mathrm{leu_{(c)}} + 0.32\,\mathrm{lys_{(c)}} + 0.24\,\mathrm{met_{(c)}} + 0.18\,\mathrm{phe_{(c)}} + 0.18\,\mathrm{pro_{(c)}} + 0.26\,\mathrm{ser_{(c)}} + 0.44\,\mathrm{thr_{(c)}} + 0.10\,\mathrm{trp_{(c)}} + 0.07\,\mathrm{tyr_{(c)}} + 0.19\,\mathrm{utp_{(c)}} + 0.32\,\mathrm{val_{(c)}} \longrightarrow 19.48\,\mathrm{adp_{(c)}} + 19.48\,\mathrm{h_{(c)}} +$$

$19.48\,\mathrm{pi_{(c)}} + 0.73\,\mathrm{ppi_{(c)}}$

Zika Virus Biomass Objective Function Equation (ZIKV_bm)
$0.38\,\mathrm{ala_{(c)}} + 0.27\,\mathrm{arg_{(c)}} + 0.14\,\mathrm{asn_{(c)}} + 0.22\,\mathrm{asp_{(c)}} + 19.93\,\mathrm{atp_{(c)}} + 0.18\,\mathrm{ctp_{(c)}} + 0.11\,\mathrm{cys_{(c)}} + 0.10\,\mathrm{gln_{(c)}} + 0.24\,\mathrm{glu_{(c)}} + 0.47\,\mathrm{gly_{(c)}} + 0.18\,\mathrm{gtp_{(c)}} + 19.75\,\mathrm{h_2o_{(c)}} + 0.14\,\mathrm{his_{(c)}} + 0.27\,\mathrm{ile_{(c)}} + 0.46\,\mathrm{leu_{(c)}} + 0.31\,\mathrm{lys_{(c)}} + 0.18\,\mathrm{met_{(c)}} + 0.15\,\mathrm{phe_{(c)}} + 0.16\,\mathrm{pro_{(c)}} + 0.33\,\mathrm{ser_{(c)}} + 0.36\,\mathrm{thr_{(c)}} + 0.10\,\mathrm{trp_{(c)}} + 0.11\,\mathrm{tyr_{(c)}} + 0.18\,\mathrm{utp_{(c)}} + 0.36\,\mathrm{val_{(c)}} \longrightarrow 19.75\,\mathrm{adp_{(c)}} + 19.75\,\mathrm{h_{(c)}} + 19.75\,\mathrm{pi_{(c)}} + 0.73\,\mathrm{ppi_{(c)}}$

Each VBOF is representative of the infected state when optimised in a metabolic network, for that particular virus. While the VBOFs include the amino acid, nucleotide and associated energy metabolite requirements they omit the presence of lipids within the virus envelopes. This is due to a lack of information regarding the ultrastructure and stoichiometry of the lipids in the virus envelopes.

The VBOFs were successfully integrated into three virus-specific human macrophage-based metabolic models: MO-CHIKV, MO-DENV and MO-ZIKV for the CHIKV, DENV and ZIKV infected metabolic networks respectively. To enable the visualisation and analysis of predicted flux values, using loopless FBA on these iHVMs, a series of visual representations of the metabolic networks were created in Escher (King et al. 2015) with mapping of flux values and directionality added manually.

2.3 RESULTS

To explore the differences in metabolic resource usage (reactions, metabolites) between the host and virus optimised systems, multiple comparisons and predictions were done for the MO-CHIKV, MO-DENV and MO-ZIKV integrated host-virus metabolic models. Stoichiometric comparisons are conducted between the three viruses regarding their biomass objective function composition. Specifically, the usage of amino acids and nucleotides is compared between the three viruses to highlight biomass-associated components that exhibit differential demands. This comparison is also applied to host-virus pairs, highlighting differential usage of amino acids and nucleotides between the host and virus objective functions. This latter analysis informs the comparison between the flux predictions for host and virus optimised states, and aids in explanation of the potential metabolic network

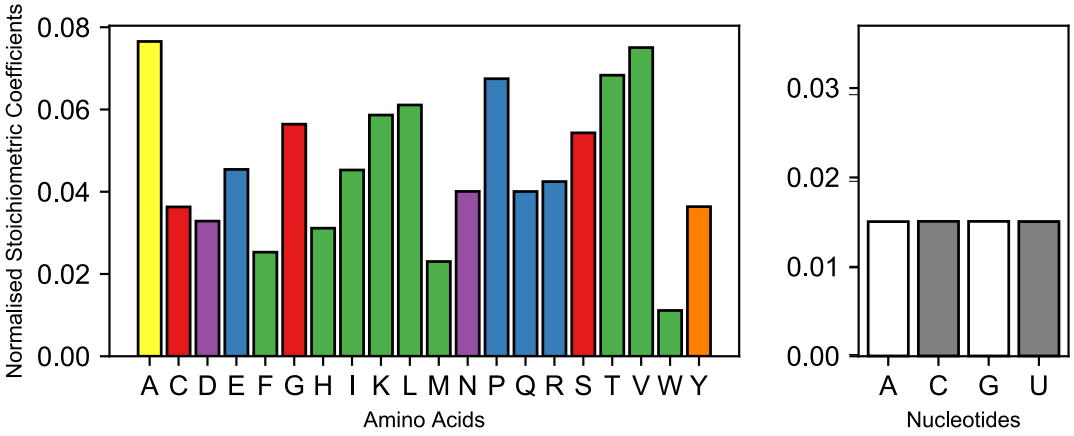
rearrangements from host-controlled to virus-controlled systems.

2.3.1 Biomass objective function compositions vary between virus species and host

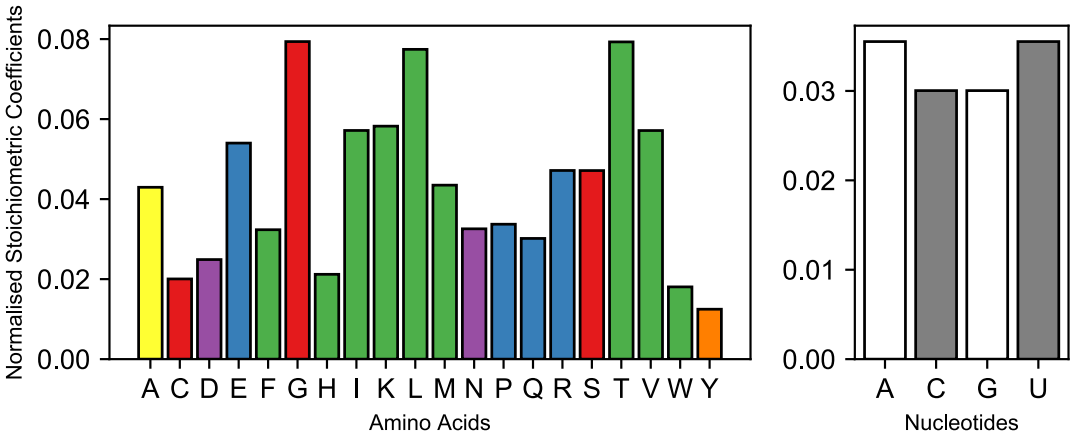
The objective functions of CHIKV, DENV and ZIKV, were compared to identify potentially critical differences in their biomass compositions. Consideration of the viral proteome, genome and viral structure (regarding protein copy number for all structural and non-structural proteins), and the virus replication cycle. This analysis highlights predicted differences in amino acid and nucleotide usage between CHIKV, DENV and ZIKV (Fig. 2.4). The three VBOFs all exhibit varying levels of amino acid and nucleotide usage. In essence, this analysis highlights how virus amino acid and nucleotide usage varies within virus genera (*Flavivirus* - DENV vs. ZIKV) and across virus genera (*Alphavirus* - CHIKV vs. *Flavivirus* - DENV and ZIKV) . The usage of amino acids can also be considered regarding the precursors they are produced from, indicated by colour coding.

For both the host and viruses (CHIKV, DENV and ZIKV), comparisons can be made between the usage of amino acids and nucleotides in their respective biomass pseudo-reactions. Quantifying this comparison requires the stoichiometric coefficients of the pseudo-reaction metabolites to be normalised and compared through log2 fold change (Fig. 2.5 and 2.6). The stoichiometric coefficients are normalised to the summation of all non-energy requirement reactant metabolites. Non-energy requirement metabolites are defined as metabolites that contribute to the biomass and are not utilised as energy sources for biomass production.

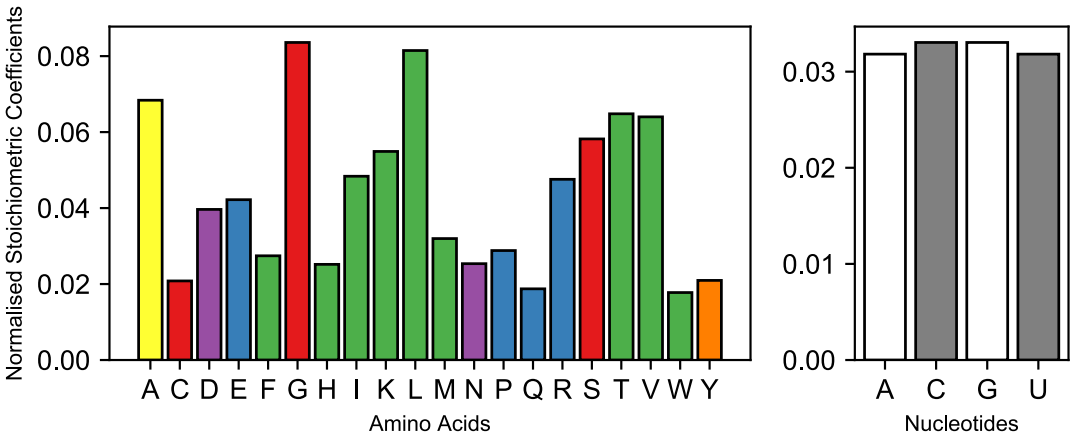
This analysis identifies amino acids or nucleotides that are used in larger (positive values) or smaller (negative values) in the virus pseudo-reaction than that of the host. Many of the amino acids show a substantial difference in their usage between the virus and host pseudo-reactions, as shown in Fig. 2.5. To relate how virus infection and composition affects the difference between uninfected and infected states, we explore the stoichiometric differences further. CHIKV has a substantial increase in the usage of cysteine when compared to the host. DENV and ZIKV both share an increased usage of methionine, whilst specifically ZIKV has increased in the use of tryptophan and glycine as well. The greatest fold-change differences for amino acids are methionine (DENV, +1.21) and glutamine (ZIKV, -1.16). These differences aid in the interpretation of the metabolic modelling results.



(a) CHIKV



(b) DENV



(c) ZIKV

Figure 2.4 ***(previous page)*****:** Amino acid and nucleotide associated normalised stoichiometric coefficients for the (a) Chikungunya, (b) Dengue and (c) Zika virus biomass objective function. The normalised (against the summation of individual virus) stoichiometric coefficient values are calculated and extracted from the virus biomass objective functions in the MO-CHIKV, MO-DENV and MO-ZIKV integrated host-virus metabolic models, respectively. Bar colour (amino acids) indicates the metabolite precursor that is associated with the biomass component: blue, 2-Oxobutanoate; green, essential amino acids (no *de novo* biosynthesis, external supply); orange, Phenylalanine; purple, Oxaloacetate; red, 3-Phosphoglycerate; yellow, Pyruvate. Common single letter abbreviations are used for amino acids and nucleotides.

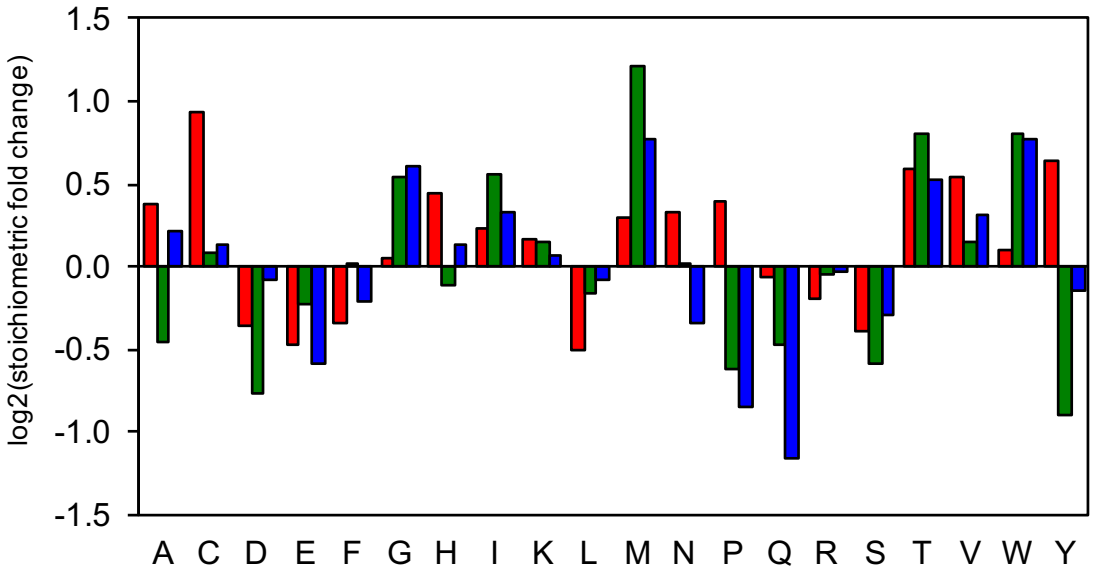


Figure 2.5: Comparison of host and virus amino acid normalised stoichiometries from their respective biomass pseudo-reactions: Chikungunya (■), Dengue (■), and Zika (■) viruses. Common single-letter abbreviations are used for amino acid metabolites.

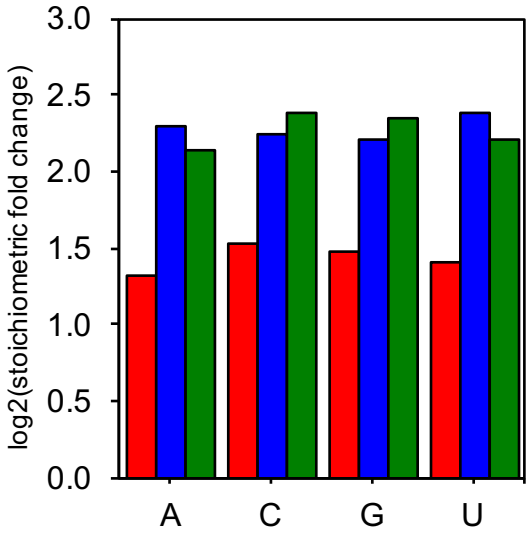


Figure 2.6: Comparison of host and virus nucleotide normalised stoichiometries from their respective biomass pseudo-reactions: Chikungunya (■), Dengue (■), and Zika (■) viruses. Common single-letter abbreviations are used for nucleotide metabolites.

Comparison between the usage of nucleotides in the biomass functions of the host and the individual viruses shows that they are considerably increased in their usage in the virus pseudo-reactions (Fig. 2.6), especially in comparison to the differences in terms of amino acid usage (Fig. 2.5); the maximum absolute difference between host and virus usage of nucleotides is 2.4 log2 fold-change compared to 1.3 log2 fold-change for amino acids. Again, this signifies that the viruses use proportionally more nucleotides for their biomass than the host. ZIKV has the most substantial difference in nucleotide usage compared to the host, with a log2 fold-change value for, cytidine and uracil, of +2.39. CHIKV

To compare the host- and virus-optimal states of the model, the metabolic fluxes directly feeding into the biomass pseudo-reactions (host and virus) were assessed. As to be expected from linear optimisation, the predicted fluxes into the biomass pseudo-reaction reflect the stoichiometric differences in the amino acid and nucleotide requirements of the host cell and the individual viruses, thus achieving perfect fulfilment of host or virus biomass requirements. Overall, however, the stoichiometric differences in metabolic requirements for virus production vs host maintenance reaction result also in different metabolic flux states in different parts of the host metabolic model.

In addition to the fulfilment of the respective biomass objective functions, the flux distributions for host- and virus-optimised states can be analysed through reaction and pathway analysis.

2.3.2 Predicted flux distributions for host and virus optimised models highlight differences in metabolite states and flows

To assess the differences in network usage, multiple reaction subsystems were individually analysed for the host and virus optimised states. These subsystems fall into three categories shown in Table 2.9.

Table 2.9: Category definitions for reaction pathways analysed

Category Description	Subsystem Name
Media Exchange	*In silico* media exchange
Central Carbon Metabolism	Glycolysis
	Pentose Phosphate Pathway
	Citric Acid Cycle
Amino Acid Biosynthesis	Serine, Glycine and Cysteine
	Aspartate and Asparagine
	Arginine and Proline

These subsystems were selected as they represent the main branches of central carbon metabolism (Almaas et al. 2005, Fell 2010) and include the *de novo* biosynthesis of nucleotides and amino acids which are essential for the host and virus biomass objective functions. We next report results from each subsystem individually, in order to evaluate the differences between host- and virus-optimised systems on a case-by-case basis.

In silico media exchange

Comparison of the model media exchange provides insights into the general state of flows in the metabolic network. Negative and positive flux values related to the directionality of a reversible reaction. In the case of exchange reactions, a negative flux value indicates the exchange metabolite is being imported, and a positive flux value indices import of the exchange metabolite. Comparison of host and virus optimised exchange reaction fluxes are shown in Fig 2.7.

All virus-optimised systems exhibit a higher import flux of oxygen (O2), glutamine and various other amino acids, along with some increases in phosphate and fatty acid (OCDCA) importation. An increase in the export of carbon dioxide and bicarbonate (CO2 and BHB) is also exhibited for all three viruses. CHIKV is the

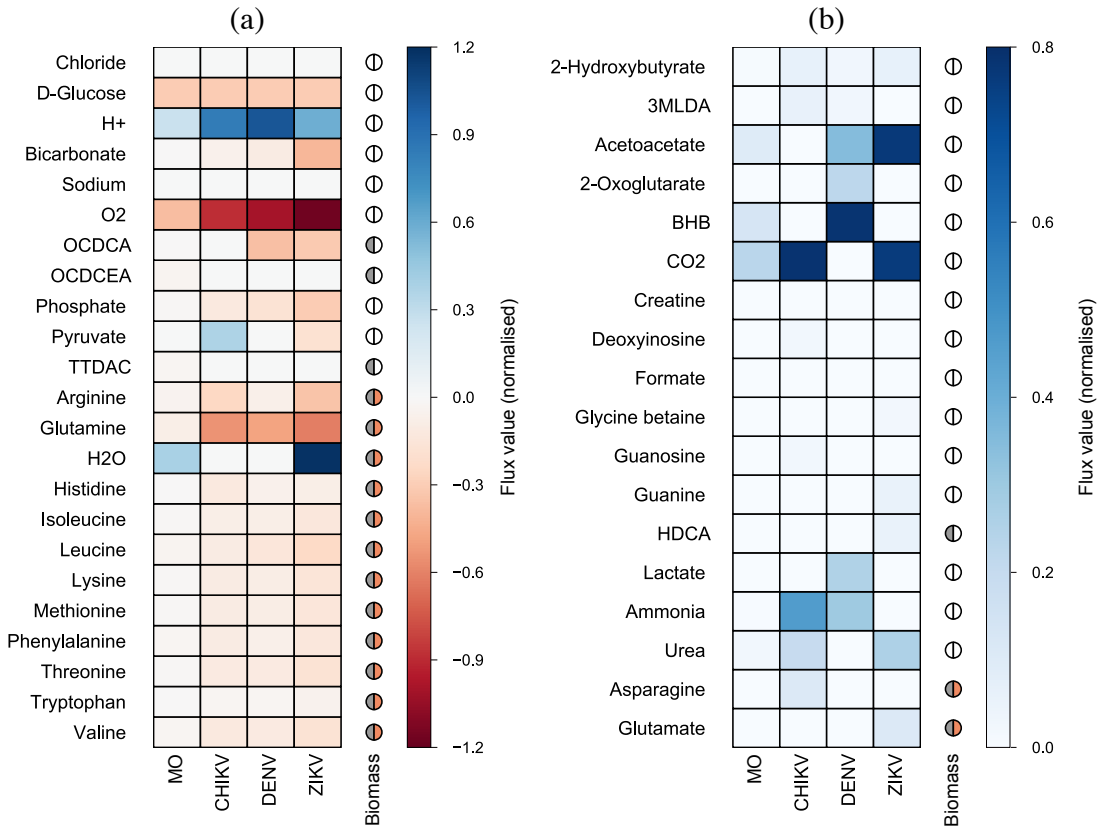


Figure 2.7: Comparison of *In silico* media exchange reaction fluxes (normalised) across host and virus optimised states. Only reactions that have a non-zero flux in at least one optimisation condition (host or virus) are shown (n = 41). Two categories of reactions shown: (a) exchange reactions that are reversible (import and export); (b) exchange reactions that are unidirectional (export only). Biomass precursor metabolites (present in the objective function) are indicated as present in host (grey-filled half-circle) and/or virus (orange-filled half-circle). Metabolite abbreviations: 3MLDA, 3-methylimidazoleacetic acid; BHB, (R)-3-hydroxybutanoate; HDCA, hexadecanoate (n-C16:0); OCDCA, octadeconate (n-C18:0); OCDCEA, octadeconate (n-C18:1); TTDCA, tetradeconate (n-C14:0). Optimisation abbreviations: MO, Host Macrophage; CHIKV, Chikungunya virus; DENV, Dengue virus; ZIKV, Zika virus.

only virus-optimised state that exhibits any flipping of import-export metabolites when compared to the host optimised system. In this case, a CHIKV optimised state results in the export of pyruvate from the metabolic model. Lactate is only produced by the DENV optimised model. There is an increased export of nitrogen-containing compounds across all virus-optimised systems (ammonia, CHIKV and DENV; urea, CHIKV and ZIKV). The export of acetoacetate is observed in both DENV and ZIKV optimised systems (as well as host), while only CHIKV and ZIKV optimised systems exhibit an increase in the export of carbon dioxide.

Overall, the virus-optimised states exhibit an increased demand for amino acid import and indeed exhibit increased import fluxes for all virus-biomass associated metabolites.

Central Carbon Metabolism

To further investigate the intracellular flux prediction differences between host and virus optimised systems, components of the previously-defined central carbon metabolism were analysed.

Glycolysis In the host optimised state (Fig. 2.8A), glycolysis runs from glucose through to pyruvate synthesis via the traditional aerobic pathway, resulting in no lactate production. Initially, glucose-6-phosphate is converted into 6PGL, used for the oxidative branch of the pentose phosphate pathway; F6P, used for continuation of glycolysis; G1P, used for glycogenesis. The production of G1P indicates that the host-optimised metabolic network is undergoing glycogenesis (the creation of glycogen). G1P activates UTP-glucose-1-phosphate uridylyltransferase and initiates the synthesis of UDP-glucose, utilised for glycogen synthesis. UDP-glucose acts as a glycosyltransferase, adding to existing glycogen molecules and liberating a UDP (produced from the initial G1P-UTP investment). The activation of glycogenesis supports existing experimental data in the analysis of activated macrophages (Galván-Peña & O'Neill 2014).

Following the continuation of the glycolytic pathway, 3-phosphoglycerate is converted to 3PHP and 2-phosphoglycerate at a ratio of 0.35:1. 3PHP is used as a precursor for serine biosynthesis (and resulting glycine and cysteine biosynthesis). Glycolysis continues through to PEP, which is supplemented by conversion from a cytosolic source of oxaloacetate as well as transport from the mitochondria, a

phenomenon seen in pancreatic β-cells (Stark et al. 2009). Finally, PEP is fully converted in a 1:1 ratio into pyruvate, which is transported to the mitochondria and converted to alanine (at a ratio of 8.73:1).

Under CHIKV optimisation, the flux distribution for the glycolytic pathway differs from that of the host optimised system (Fig. 2.8A and B). Glycolysis only proceeds to 3PG, where there is then full conversion to 3PHP and commitment to the serine biosynthesis pathway. Unlike in the host optimised system, there is no glycogen synthesis via the conversion of G6P to G1P. However, G6P is converted to 6PGL indicating the use of the oxidative branch of the pentose phosphate pathway. Pyruvate is synthesised via mitochondrial-sourced PEP but also through the conversion of DHAP via methylglyoxal synthase (MGSA), eventually forming methylglyoxal (the reduced penultimate intermediate to pyruvate). Pyruvate is then exclusively utilised in alanine biosynthesis with the remainder exported to the extracellular compartment. There is no cytosol-mitochondria transport of pyruvate.

DENV optimisation predicts that glycolysis (Fig. 2.8A and C), like CHIKV, runs through to 3PG synthesis, where there is the full conversion to 3PHP synthesis and the serine biosynthetic pathway. However, in the upper stages of glycolysis, G6P is not converted to 6PGL indicating that there is no activation of the non-oxidative branch of the pentose phosphate pathway. Instead, glycolytic intermediates F6P and G3P are utilised for the non-oxidative branch of the pentose phosphate pathway. Again, in a similar manner to CHIKV, DHAP is not solely converted to G3P but instead is also metabolised to pyruvate via MGSA, with a higher flux value for the DENV-optimised system (0.19 mmol glucose^{-1} h^{-1}) than the CHIKV-optimised system (0.41 mmol glucose^{-1} h^{-1}). Lower glycolysis is supplemented via mitochondrial-sourced PEP which, along with the byproducts of methylglyoxal metabolism, drives the synthesis of pyruvate. Lactate dehydrogenase is active in the DENV optimised system, resulting in the production of lactate from pyruvate (-0.65 mmol glucose^{-1} h^{-1}) which is exported to the extracellular compartment. Pyruvate is also utilised for alanine biosynthesis, at a ratio of 1:0.15 (alanine:lactate). There is no cytosol-mitochondria transport of pyruvate.

Finally, the optimisation of ZIKV predicts that glycolysis only runs from glucose to the first glycolytic intermediate G6P (Fig. 2.8A and D). There is the full commitment (1 mmol glucose^{-1} h^{-1}) of G6P to the production of 6PGL, and thus glucose is fully utilised for the oxidative branch of the pentose phosphate path-

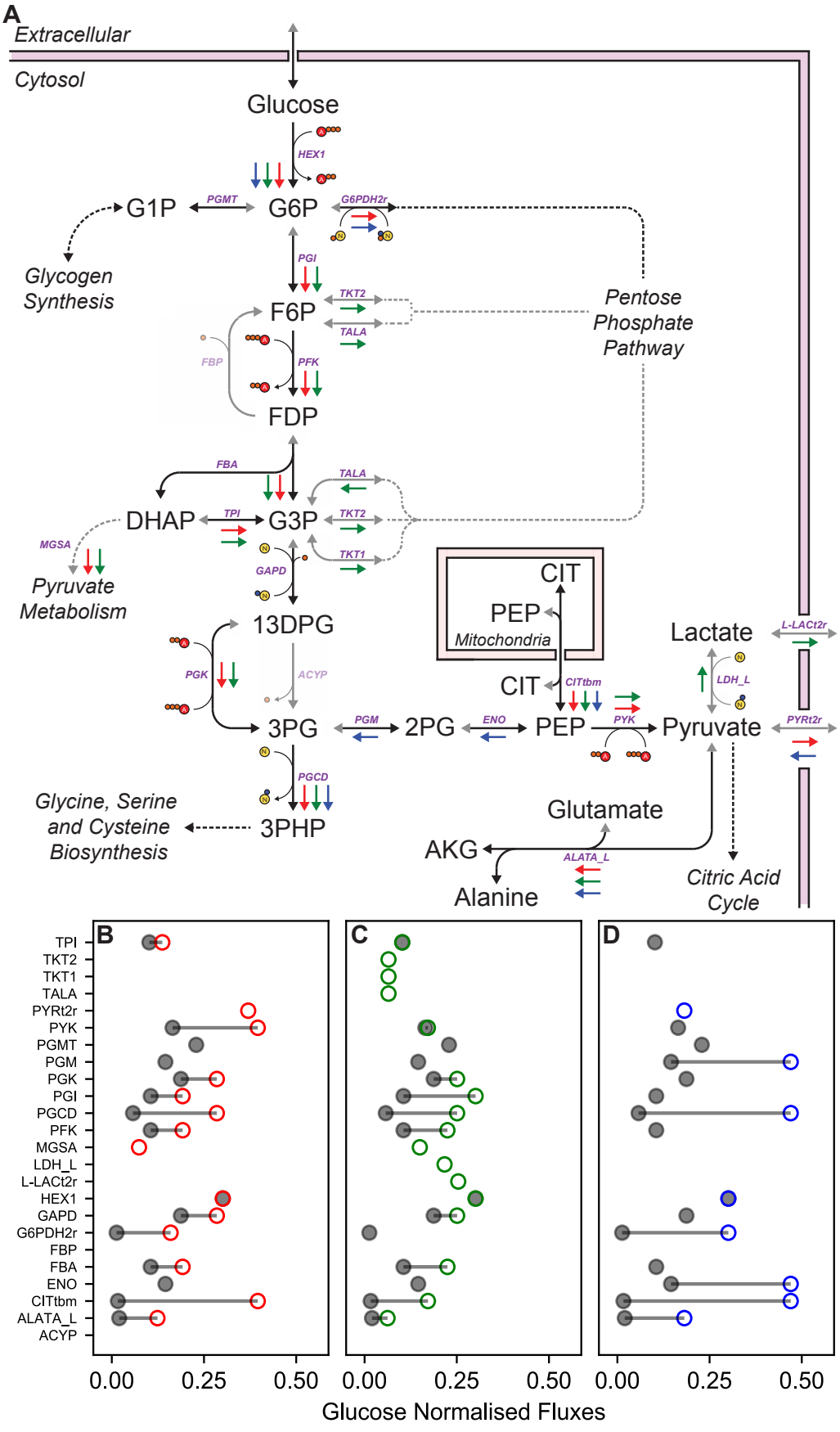

A
Extracellular
Cytosol
Glucose
HEX1
G1P
PGMT
G6P
G6PDH2r
Glycogen Synthesis
PGI
F6P
TKT2
TALA
Pentose Phosphate Pathway
FBP
PFK
FDP
FBA
TALA
DHAP
TPI
G3P
TKT2
MGSA
GAPD
TKT1
Pyruvate Metabolism
13DPG
CIT
PEP
Mitochondria
Lactate
L-LACt2r
PGK
ACYP
CIT
CITtbm
LDH_L
3PG
PGM
2PG
ENO
PEP
PYK
Pyruvate
PYRt2r
PGCD
Glycine, Serine and Cysteine Biosynthesis
3PHP
Glutamate
AKG
ALATA_L
Alanine
Citric Acid Cycle
B
C
D
TPI
TKT2
TKT1
TALA
PYRt2r
PYK
PGMT
PGM
PGK
PGI
PGCD
PFK
MGSA
LDH_L
L-LACt2r
HEX1
GAPD
G6PDH2r
FBP
FBA
ENO
CITtbm
ALATA_L
ACYP
0.00
0.25
0.50
Glucose Normalised Fluxes

Figure 2.8 ***(previous page)*:** Glycolysis metabolic pathway map (**A**) with comparison of predicted host- and virus- optimised flux values (loopless FBA) for (**B**) CHIKV, (**C**) DENV, and (**D**) ZIKV host-virus integrated metabolic models. Arrows indicate reactions, arrowheads indicate potential directionality (two arrowheads indicate a bi-directional reaction). Dashed arrows indicate a connecting set of reactions or pathway where flux has not been explicitly indicated. Host fluxes are indicated through black arrows (➔) with directionality indicated by black arrowhead (grey arrows ➔ and arrow heads indicate no usage of reaction or directionality). Virus optimisations and directionality are indicated by coloured arrows for CHIKV (➔), DENV (➔) and ZIKV (➔). Virus optimised fluxes are compared to host optimised fluxes through normalisation against glucose. Optimised fluxes are indicated through circle markers: Host (●); CHIKV (○); DENV (○); ZIKV (○). Full reaction and metabolite names are given in Appendix B and C, respectively. CHIKV, Chikungunya virus; DENV, Dengue virus; ZIKV, Zika virus.

way. Unlike the CHIKV and DENV optimised systems, ZIKV has some reversed segments of the lower glycolytic pathways. Specifically, mitochondrial PEP drives the synthesis of 3PG (via 2PG) which is then exclusively used for the production of 3PHP for the serine biosynthetic pathway. Pyruvate is imported from the extracellular environment (-1.95 mmol glucose^{-1} h^{-1}), and subsequently only used for alanine biosynthesis; there is no lactate production or cytosol-mitochondria transport. There is also no activation of methylglyoxal associated pathways or glycogen synthesis.

Overall, ZIKV is predicted to have the largest allocation of flux to both serine (PGCD) and alanine (ALATA_L) biosynthesis, with flux values of 1.95 mmol glucose^{-1} h^{-1}and -0.51 mmol glucose^{-1} h^{-1}respectively. For comparison, the values for the other viruses are approximately half of ZIKV: CHIKV, PGCD (0.92 mmol glucose^{-1} h^{-1}) and ALATA_L (-0.33 mmol glucose^{-1} h^{-1}); DENV, PGCD (0.92 mmol glucose^{-1} h^{-1}) and ALATA_L (0.78 mmol glucose^{-1} h^{-1}).

Pentose Phosphate Pathway The host optimised system predicts activation of the oxidative branch of the pentose phosphate pathway (Fig. 2.9 A). There is the full conversion of 6PGl (G6P-derived) to R5P. The synthesis of R5P is supplemented via nucleotide salvage pathways, specifically by adenosine salvage (purine-nucleoside phosphorylase PUNP1) to produce the R5P precursor R1P. R5P is then converted to PRPP, which is subsequently used for purine and pyrimidine nucleotide biosynthesis with a ratio of 1.03:1. There is also some diversion of PRPP

to be utilised as a co-factor for the conversion of guanine (deoxyguanine derived) to GMP (via GUARPT) ultimately generating ribose-purines from deoxyribose-purine salvage.

Metabolic analysis predicts that CHIKV and ZIKV qualitatively utilise the pentose phosphate pathway similarly to the host (Fig. 2.9B and D), and each other, although the absolute flux through ZIKV-active reactions is higher than that of the CHIKV-active counterparts. This is due to the exhaustive use of glucose to drive the pentose phosphate pathway in the ZIKV-optimised system: the flux through the initial step of the pentose phosphate pathway for ZIKV compared to CHIKV is 1 mmol glucose^{-1} h^{-1}and 0.44 mmol glucose^{-1} h^{-1}respectively. Both viruses utilise the oxidative branch to produce PRPP, which proceeds on to purine and pyrimidine nucleotide biosynthesis at ratios of 1.83:1 (CHIKV) and 1.29:1 (ZIKV). However, in the CHIKV-optimised system there is no recruitment of nucleotide salvage pathways, whereas, for the ZIKV-optimised system, guanosine (via purine-nucleoside phosphorylase PUNP3) is utilised as a catabolic reactant (rather than adenosine in the host). Nucleotide salvage pathways (via PPM) supplement the production of nucleotides with a flux of 0.14 mmol glucose^{-1} h^{-1}, much higher than the host-optimised flux of 0.0025 mmol glucose^{-1} h^{-1}.

DENV, unlike the host-, CHIKV- or ZIKV-optimised systems, utilises the non-oxidative branch of the pentose phosphate pathway (Fig. 2.9A and C). F6P and G3P are utilised as products to produce XU5P-D and R5P, the former of which is converted into R5P via the RU5P-D intermediate. There is no activation of nucleotide salvage pathways, and R5P is converted into PRPPS which is utilised for purine and pyrimidine biosynthesis in a 1:1 ratio. DENV exhibits similar reaction fluxes through the final steps of the pentose phosphate pathway, phosphatidylserine decarboxylase (PRPPS), as CHIKV (0.48 and 0.44 mmol glucose^{-1} h^{-1}, respectively). The ZIKV-optimised flux of PRPPS, however, is $2.4 - 2.6$x the flux of CHIKV and DENV, at 1.15 mmol glucose^{-1} h^{-1}. This flux includes the entire contribution of the *in silico* glucose to the pentose phosphate pathway and supplementation via the guanosine-nucleotide salvage pathways.

Citric Acid Cycle

Under host control, the citric acid cycle (CAC) is predominantly run via cytosol-sourced pyruvate produced via glycolysis (Fig. 2.10A). Pyruvate is used to regener-

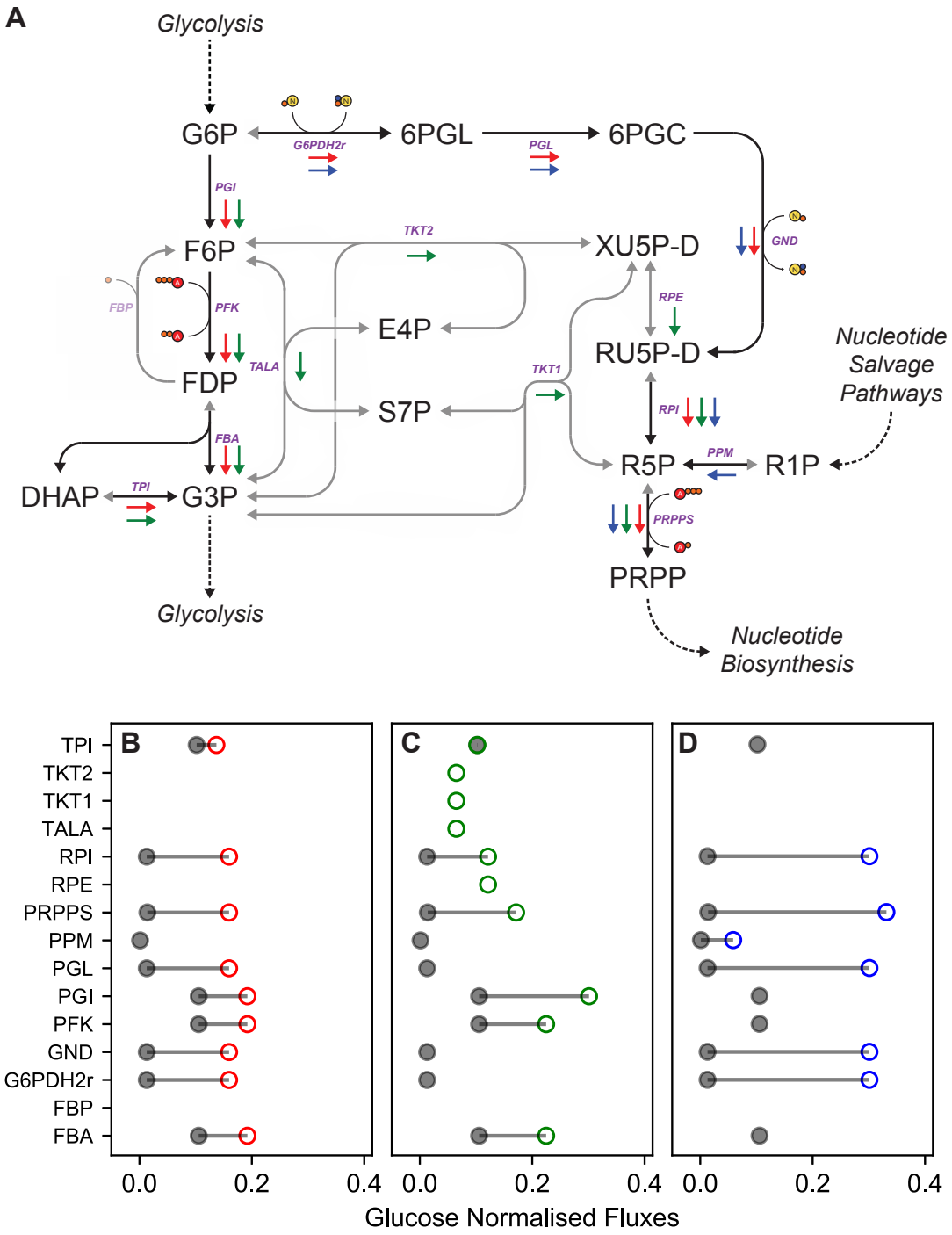


Figure 2.9: Pentose Phosphate Pathway metabolic pathway map (**A**) with comparison of predicted host- and virus- optimised flux values (loopless FBA) for (**B**) CHIKV, (**C**) DENV, and (**D**) ZIKV host-virus integrated metabolic models. Arrows indicate reactions, arrowheads indicate potential directionality (two arrowheads indicate a bi-directional reaction). Dashed arrows indicate a connecting set of reactions or pathway where flux has not been explicitly indicated.
(continued on next page...)

Figure 2.9 ***(previous page)*:** Host fluxes are indicated through black arrows (➔) with directionality indicated by black arrowhead (grey arrows ➔ and arrow heads indicate no usage of reaction or directionality). Virus optimisations and directionality are indicated by coloured arrows for CHIKV (➔), DENV (➔) and ZIKV (➔). Virus optimised fluxes are compared to host optimised fluxes through normalisation against glucose. Optimised fluxes are indicated through circle markers: Host (●); CHIKV (○); DENV (○); ZIKV (○). Full reaction and metabolite names are given in Appendix B and C, respectively. CHIKV, Chikungunya virus; DENV, Dengue virus; ZIKV, Zika virus.

ate citrate (CIT) from oxaloacetate (OAA) using acetyl coenzyme A (ACCOA) as a cofactor. Complete running of the CAC involves the conversion of CIT through to isocitrate (ICIT), however, under host control, CIT is used as a co-factor for the antiport co-transportation of phosphoenolpyruvate (PEP) and malate (MAL) in the mitochondria-cytosol and cytosol-mitochondria direction, respectively. As the conversion of CIT to ICIT and α-ketoglutarate (AKG) does not occur, instead, OAA is converted to AKG and aspartate using glutamate as a co-factor. AKG is then consumed to produce glutamate via glutamate dehydrogenase and ornithine transaminase; the latter is associated with the urea cycle. Glutaminolysis is also used to drive the mitochondrial biosynthesis of glutamate via mitochondrial glutaminase (GLUNm). The CAC then continues through succinate-CoA (SUCCOA), which is supplemented by byproducts of fatty acid metabolism and ketone body metabolism, and succinate (SUCC). CAC then continues through fumarate (FUM) and malate (MAL), both metabolites are supplemented by FUM and MAL importation from the cytosol, to produce OAA. OAA is then used to drive CIT, and AKG/aspartate synthesis at a ratio of 1.87:1. Finally, the remaining OAA is utilised for PEP production, which is transported out of the mitochondria to the cytosol and used to drive the lower end of the glycolytic pathway.

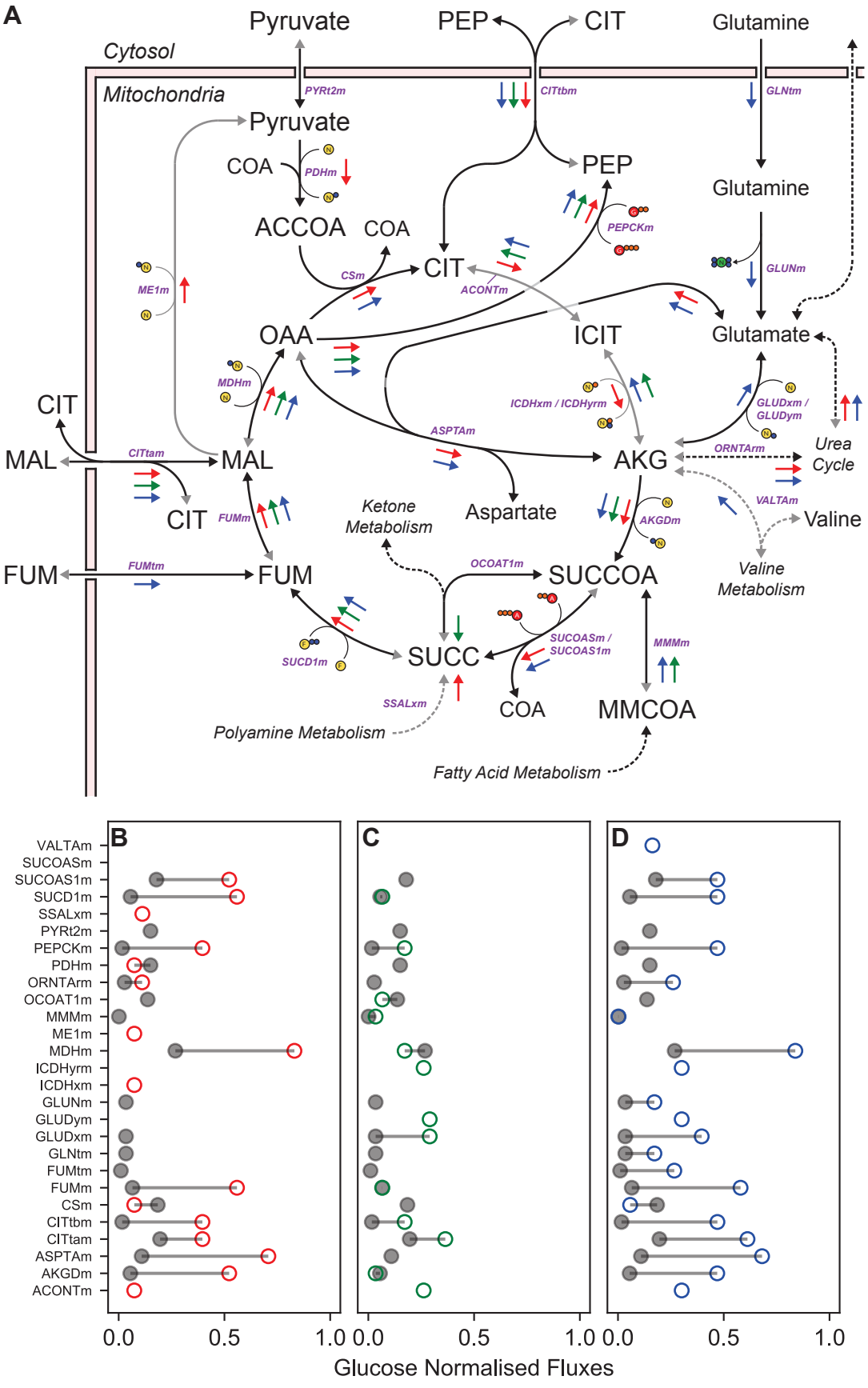

A
Pyruvate
PEP
CIT
Glutamine
Cytosol
Mitochondria
PYRt2m
CITtbm
GLNtm
Pyruvate
COA
PDHm
PEP
Glutamine
ACCOA
COA
PEPCKm
CSm
CIT
ACONTm
GLUNm
ME1m
OAA
ICIT
Glutamate
MDHm
ICDHxm / ICDHyrm
GLUDxm / GLUDym
CIT
ASPTAm
ORNTArm
Urea Cycle
MAL
CITtam
MAL
AKG
CIT
FUMm
Ketone Metabolism
Aspartate
AKGDm
VALTAm
Valine
FUM
FUMtm
FUM
OCOAT1m
SUCCOA
Valine Metabolism
SUCOASm / SUCOAS1m
MMMm
SUCD1m
SUCC
SSALxm
COA
MMCOA
Polyamine Metabolism
Fatty Acid Metabolism
B
C
D
VALTAm
SUCOASm
SUCOAS1m
SUCD1m
SSALxm
PYRt2m
PEPCKm
PDHm
ORNTArm
OCOAT1m
MMMm
ME1m
MDHm
ICDHyrm
ICDHxm
GLUNm
GLUDym
GLUDxm
GLNtm
FUMtm
FUMm
CSm
CITtbm
CITtam
ASPTAm
AKGDm
ACONTm
0.0
0.5
1.0
Glucose Normalised Fluxes

Figure 2.10 ***(previous page)*****:** Citric Acid Cycle metabolic pathway map (**A**) with comparison of predicted host- and virus- optimised flux values (loopless FBA) for (**B**) CHIKV, (**C**) DENV, and (**D**) ZIKV host-virus integrated metabolic models. Arrows indicate reactions, arrowheads indicate potential directionality (two arrowheads indicate a bi-directional reaction). Dashed arrows indicate a connecting set of reactions or pathway where flux has not been explicitly indicated. Host fluxes are indicated through black arrows (➔) with directionality indicated by black arrowhead (grey arrows ➔ and arrow heads indicate no usage of reaction or directionality). Virus optimisations and directionality are indicated by coloured arrows for CHIKV (➔), DENV (➔) and ZIKV (➔). Virus optimised fluxes are compared to host optimised fluxes through normalisation against glucose. Optimised fluxes are indicated through circle markers: Host (●); CHIKV (○); DENV (○); ZIKV (○). Full reaction and metabolite names are given in Appendix B and C, respectively. CHIKV, Chikungunya virus; DENV, Dengue virus; ZIKV, Zika virus.

In contrast to the host optimised system, none of the CHIKV-, DENV- or ZIKV-optimised systems utilises cytosolic-sourced pyruvate for the running of the CAC; they all exhibit decoupling of the CAC from the glycolytic pathway and associated metabolic intermediates. However, CHIKV is the only virus (and indeed the only optimised system) that has a full complete run through of the CAC from CIT to OAA (Fig. 2.10A and B). In a similar manner to the host, CHIKV utilises cytosolic CIT for the antiport co-transportation of PEP (which is used to drive the synthesis of final stage glycolytic intermediates). CIT is also converted via ICIT to AKG. Glutamate synthesis occurs purely from AKG via the urea cycle reactions and as a byproduct of polyamine metabolism; there is no activation of glutamate dehydrogenase. Glutamate is not supplemented via Glutamine degradation (Glutaminolysis). Glutamate is utilised in conjunction with OAA for the synthesis of aspartate (and AKG). Continuing the CAC, SUCCOA is converted through to SUCC, the latter of which is supplemented by the metabolism of polyamines which requires the proteogenic amino acid methionine and non-proteogenic amino acid ornithine. The latter is only produced via the urea cycle, whilst methionine is extracellularly sourced. SUCC is then converted through to FUM and MAL. MAL is supplemented with antiport co-transportation from the cytosol, with CIT as a co-factor. MAL is then used in the replenishment of OAA, but also the synthesis of mitochondrial pyruvate via malic enzyme (ME1m). This pyruvate is then utilised to produce CIT from OAA, which is also used for aspartate and AKG production, at a ratio of 0.045:1. This ratio is much lower than that of the host, indicating a

higher proportion of CAC flux that is diverted to aspartate (ASPTAm) and AKG synthesis.

The DENV-optimised system also exhibits CAC decoupling from glycolysis due to the lack of pyruvate cytosolic importation (Fig. 2.10A and C). However, under DENV-optimisation the CAC does not run entirely from CIT to OAA. Instead, CIT has generated from the reversal of intermediate components of the CAC; Isocitrate dehydrogenase (ICDHxm/ICDHyrm), converted AKG to CIT via ICIT. CIT is then, as is the case with host and virus optimised systems, utilised as a co-factor in the antiport co-transportation of MAL and PEP. In place of CIT-derived AKG synthesis, AKG is imported from the cytosol using MAL as a co-factor (reaction not shown). AKG is then used exclusively to drive the CAC for the synthesis of CIT and SUCCOA. There, AKG is utilised for glutamate biosynthesis or activation of the urea cycle. SUCCOA associated reactions are supplemented by flux from fatty acid metabolism, and SUCCOA is then metabolised to SUCC via conversion through the additional co-factors derived from ketone body metabolism, rather than the CAC associated Succinate-CoA ligase (SUCOASm/SUCOAS1m). The CAC then continues through FUM and MAL, again supplemented by cytosolic importation, and eventually to OAA. OAA is then used exclusively for PEP synthesis; OAA is not utilised as a co-factor in the interconversion of glutamate and aspartate. PEP is then transported via antiport to facilitate the synthesis of cytosolic pyruvate via glycolysis.

Both the ZIKV and DENV optimised systems do not have a sufficiently completed CAC cycle, from CIT to OAA, and both (along with CHIKV) are considered decoupled from glycolysis. However, in the ZIKV optimised system there is activation of the anaplerotic pathways that drive glutaminolysis, identical to the pathways activated in the host optimised system (Fig. 2.10A and D); the glutamine cytosol-mitochondria flux is 6-times higher in the ZIKV- (0.49 mmol glucose^{-1} h^{-1}) than host- (0.08 mmol glucose^{-1} h^{-1}) optimised system. This anaplerotic flux is then used to drive the synthesis of glutamate which ultimately is interconverted to aspartate (utilising OAA and producing AKG). Similar to DENV, AKG is then used to drive the synthesis of SUCCOA and the reverse CAC synthesis of CIT. SUCCOA is supplemented by the metabolism of fatty acids and drives through to the regeneration of OAA via MAL and FUM, both of which are again supplemented by cytosolic import. As with CHIKV, OAA is utilised for CIT, aspartate/AKG and PEP synthesis. However, in the case of the latter, whilst PEP is transported to the cytosol it is not used for pyruvate synthesis but instead drives the reversal

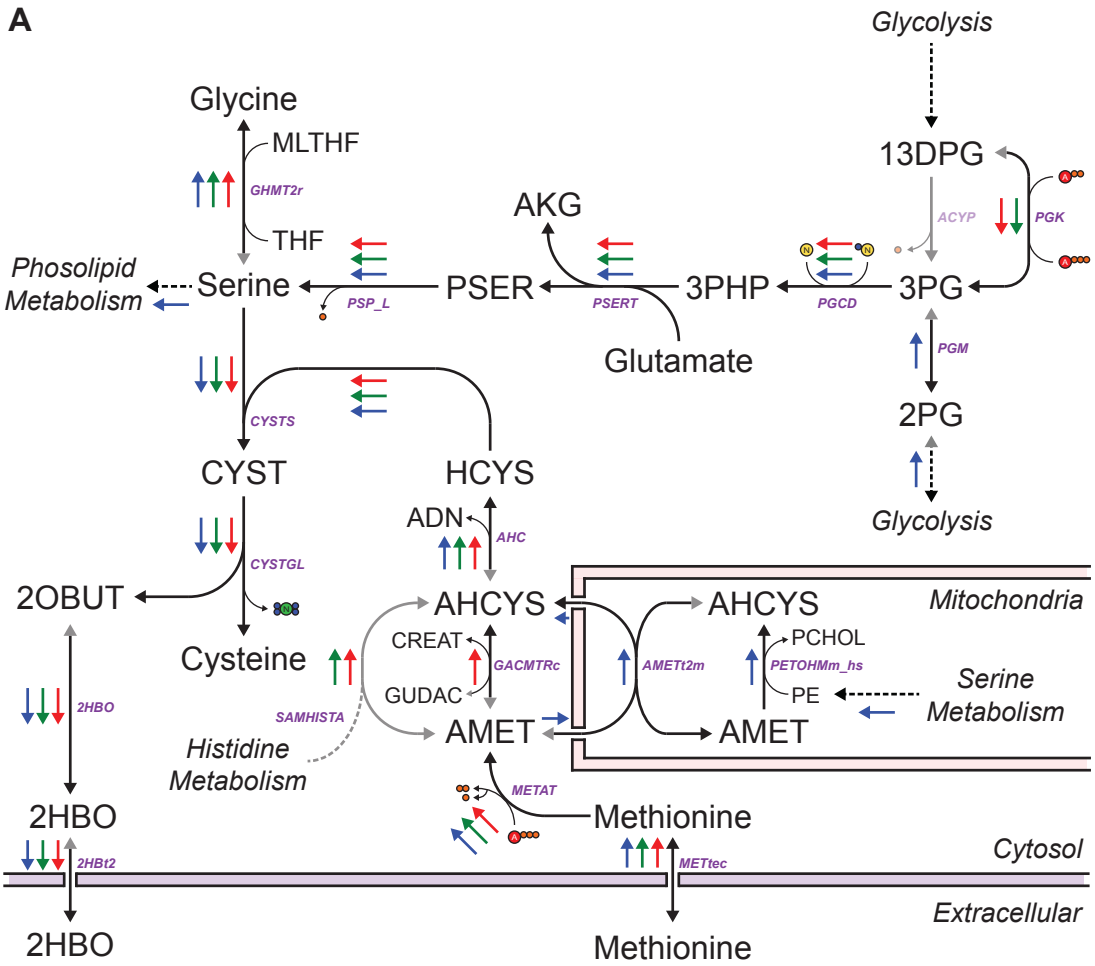
of glycolysis.

Amino Acid Biosynthesis

As previously discussed in the glycolysis pathway, under the host optimised system the glycolytic intermediate 3PG drives both glycolytic and serine biosynthetic flux at a ratio of 2.82:1 (Fig 2.11A). For the latter, 3PG is converted through to serine. Serine is a biomass component of the host but is also used for the biosynthesis of glycine and cysteine, as well as phospholipid metabolism. Specifically, serine is used for the synthesis of sphingosine and phosphatidylserine, both of which are utilised in cell membranes and are precursors for the host biomass pseudo-reaction. The synthesis of cysteine from serine requires homo-cysteine (HCYS), which is derived from the methionine catabolic product S-adenosyl-L-methionine (AMET). AMET is converted to the penultimate HCYS precursor S-adenosyl-L-homocysteine (AHCYS) via cytosolic and mitochondrial (including transport) methods. Cytosolic AMET to AHCYS metabolism requires the investment of guanidinoacetate (GUDAC), achieved through the modification of the glycine amino group, which liberates creatine (CREAT) in the process. The mitochondrial conversion requires the serine-derived phospholipid metabolic product phosphatidylethanolamine (PE). Ultimately cysteine is synthesised, which results in the production of the byproduct 2-Oxobutanoate (2OBUT), which is metabolised and exported to the extracellular compartment.

Serine, Glycine and Cysteine. Unlike the host optimised system, the virus optimised systems utilise all of the glycolytic intermediate 3PG for the biosynthesis of serine. However, for CHIKV and DENV (Fig 2.11B and C) 3PG is entirely glucose-derived, whereas in ZIKV (Fig 2.11D) it is produced by reversed reactions in lower glycolysis, and driven by PEP-derived metabolites. In all virus conditions, 3PG is converted through to serine, where it is then utilised for the biosynthesis of glycine and cysteine; only in the case of ZIKV is the phospholipid metabolism pathway active. Compared to the host all three viruses exhibit a much higher serine-glycine biosynthetic [glucose normalised] flux (GHMT2r): a nine fold-increase for CHIKV and DENV but a 21 fold-increase for ZIKV. In the case of ZIKV, glycine represents a much more significant biomass component than that of CHIKV and DENV (Fig. 2.4) and that of the host (Fig. 2.5).

Cysteine is synthesised from serine, for all three viruses, and as in the case of the


A
Glycolysis
Glycine
MLTHF
GHMT2r
THF
13DPG
ACYP
PGK
AKG
Phosolipid Metabolism
Serine
PSP_L
PSER
PSERT
3PHP
PGCD
3PG
Glutamate
PGM
2PG
CYSTS
CYST
HCYS
Glycolysis
ADN
AHC
CYSTGL
2OBUT
AHCYS
AHCYS
Mitochondria
CREAT
GACMTRc
AMETt2m
PCHOL
PETOHMm_hs
PE
Serine Metabolism
Cysteine
GUDAC
2HBO
SAMHISTA
AMET
AMET
Histidine Metabolism
METAT
Methionine
2HBO
2HBt2
METtec
Cytosol
Extracellular
2HBO
Methionine

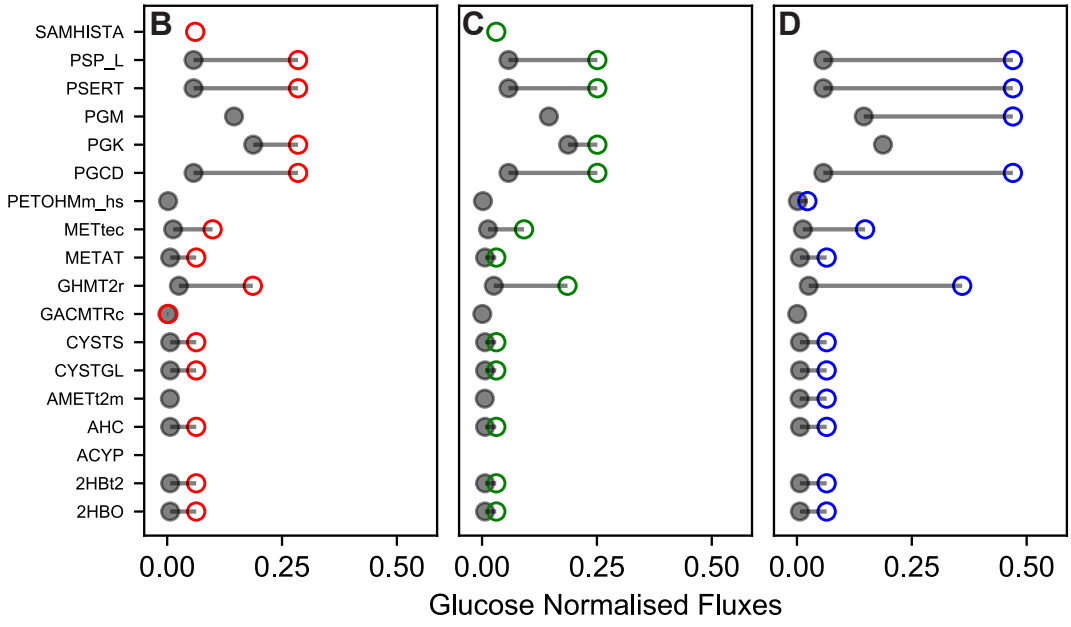

B
C
D
SAMHISTA
PSP_L
PSERT
PGM
PGK
PGCD
PETOHMm_hs
METtec
METAT
GHMT2r
GACMTRc
CYSTS
CYSTGL
AMETt2m
AHC
ACYP
2HBt2
2HBO
0.00
0.25
0.50
Glucose Normalised Fluxes

Figure 2.11 ***(previous page)*:** Serine, Glycine and Cysteine biosynthesis metabolic pathway map (**A**) with comparison of predicted host- and virus- optimised flux values (loopless FBA) for (**B**) CHIKV, (**C**) DENV, and (**D**) ZIKV host-virus integrated metabolic models. Arrows indicate reactions, arrowheads indicate potential directionality (two arrowheads indicate a bi-directional reaction). Dashed arrows indicate a connecting set of reactions or pathway where flux has not been explicitly indicated. Host fluxes are indicated through black arrows (➔) with directionality indicated by black arrowhead (grey arrows ➔ and arrow heads indicate no usage of reaction or directionality). Virus optimisations and directionality are indicated by coloured arrows for CHIKV (➔), DENV (➔) and ZIKV (➔). Virus optimised fluxes are compared to host optimised fluxes through normalisation against glucose. Optimised fluxes are indicated through circle markers: Host (●); CHIKV (○); DENV (○); ZIKV (○). Full reaction and metabolite names are given in Appendix B and C, respectively. CHIKV, Chikungunya virus; DENV, Dengue virus; ZIKV, Zika virus.

host also requires the synthesis of the penultimate cysteine precursor HCYS. However, only CHIKV exhibits a similar state to the host optimised system through the activation of GUDAC-dependent AHCYS synthesis. However, both CHIKV and DENV utilise non-mitochondrial routes for AHCYS synthesis, although DENV exclusively utilises the histidine-based S-adenosyl-L-methionine:histamine N-telemethyltransferase (SAMHISTA) reaction, and for CHIKV the histidine-based pathway carries a substantially larger flux than that of the GUDAC-based pathway, a ratio of 26.6:1. ZIKV exclusively utilises mitochondrial conversion of AMET to AHCYS, which requires the usage of serine-derived phospholipids as co-factors (similar to the host optimised state).

Aspartate and Asparagine. For the host, CHIKV and DENV optimised systems, aspartate is synthesised from mitochondrial OAA and transported (via glutamate antiport) into the cytosol (Fig 2.12 A, B and C). No mitochondrial aspartate biosynthesis occurs for the DENV-optimised system; aspartate is synthesised from cytosolic sources of OAA. In the host- and virus-optimised systems, aspartate is utilised for biomass production, nucleotide synthesis and the biosynthesis of asparagine (ASNS1).

In the case of CHIKV, there is a much higher aspartate and asparagine biosynthetic flux compared to the host: 14 and 18 fold-increase respectively. Whilst aspartate and asparagine are used in the CHIKV biomass; asparagine is also used as a co-factor for the co-transport (via antiport) of extracellular threonine into the

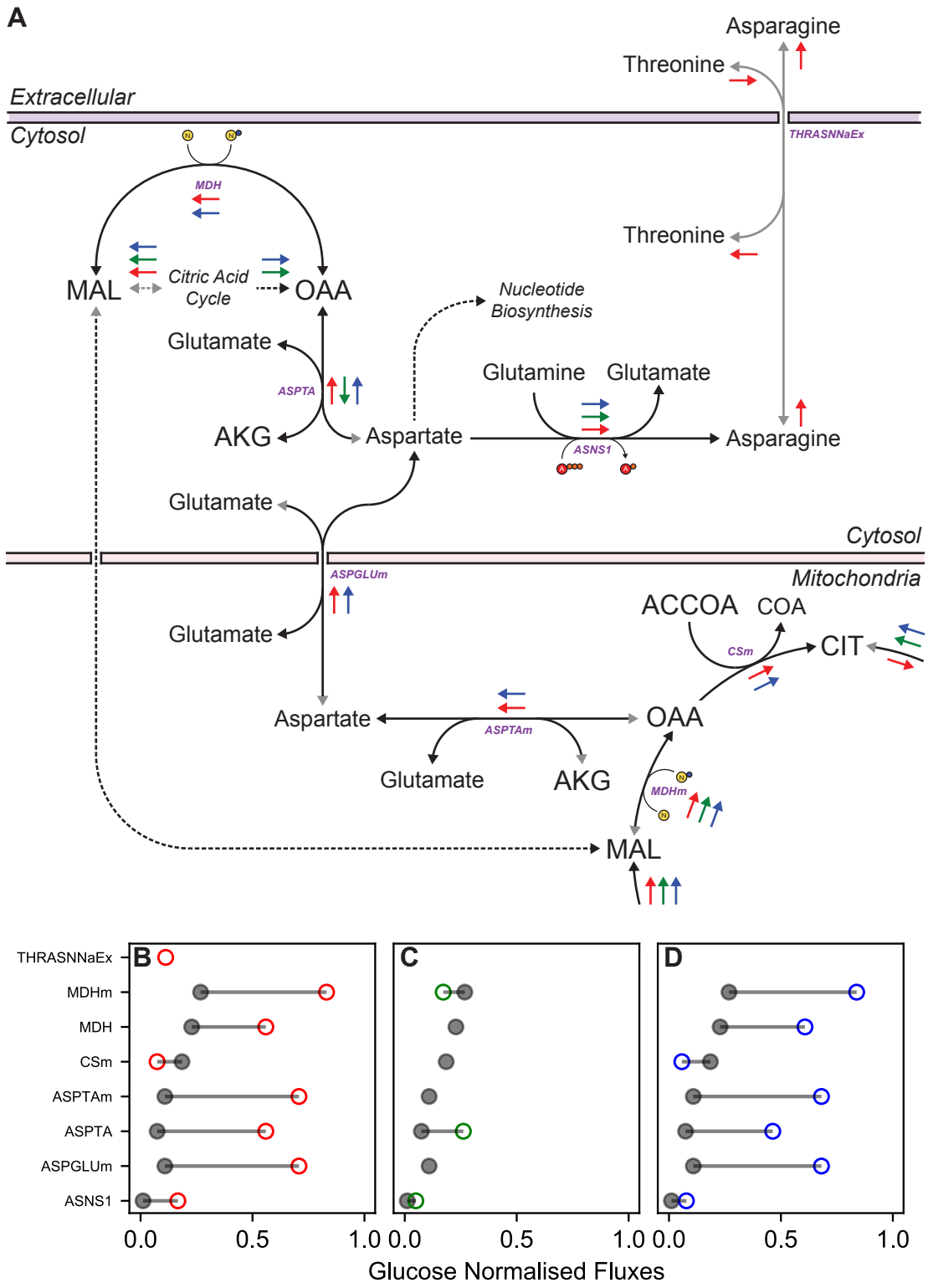

A
Asparagine
Threonine
Extracellular
Cytosol
THRASNNaEx
MDH
Threonine
MAL
Citric Acid Cycle
OAA
Nucleotide Biosynthesis
Glutamate
ASPTA
Glutamine
Glutamate
AKG
Aspartate
ASNS1
Asparagine
Glutamate
Cytosol
ASPGLUm
Mitochondria
ACCOA
COA
Glutamate
CSm
CIT
Aspartate
ASPTAm
OAA
Glutamate
AKG
MDHm
MAL
THRASNNaEx
MDHm
MDH
CSm
ASPTAm
ASPTA
ASPGLUm
ASNS1
B
C
D
0.0
0.5
1.0
Glucose Normalised Fluxes

Figure 2.12 ***(previous page)*:** Aspartate and Asparagine biosynthesis metabolic pathway map (**A**) with comparison of predicted host- and virus- optimised flux values (loopless FBA) for (**B**) CHIKV, (**C**) DENV, and (**D**) ZIKV host-virus integrated metabolic models. Arrows indicate reactions, arrowheads indicate potential directionality (two arrowheads indicate a bi-directional reaction). Dashed arrows indicate a connecting set of reactions or pathway where flux has not been explicitly indicated. Host fluxes are indicated through black arrows (➔) with directionality indicated by black arrowhead (grey arrows ➔ and arrow heads indicate no usage of reaction or directionality). Virus optimisations and directionality are indicated by coloured arrows for CHIKV (➔), DENV (➔) and ZIKV (➔). Virus optimised fluxes are compared to host optimised fluxes through normalisation against glucose. Optimised fluxes are indicated through circle markers: Host (●); CHIKV (○); DENV (○); ZIKV (○). Full reaction and metabolite names are given in Appendix B and C, respectively. CHIKV, Chikungunya virus; DENV, Dengue virus; ZIKV, Zika virus.

cytosol. ZIKV also exhibits a higher aspartate and asparagine biosynthetic flux compared to host (13 and eight fold-increase) although there is no activation of the asparagine-threonine antiport exchange (Fig 2.12A and D). Both CHIKV and ZIKV utilise aspartate as a source for OAA production, ultimately driving the cytosolic production of MAL which is then transported to the mitochondria and used to feed the CAC. Whilst DENV also transports MAL across the mitochondria for CAC, the source of OAA is not aspartate-derived.

Arginine and Proline

Host, CHIKV and ZIKV utilisation of the urea cycle (Fig 2.13 A, B and D), for the utilisation of arginine and production of proline, requires the importation of arginine from the extracellular compartment. In the case of DENV (Fig 2.13 C), the arginine that is imported is only utilised in the biomass pseudo-reaction; there is no catabolism of arginine for any other metabolic processes. Host, CHIKV and ZIKV convert the arginine to ornithine (ORN). Here, ORN is transported to the mitochondria, but in the case of CHIKV, it is also metabolised to putrescine (PTRC). Ultimately, PTRC is metabolised further and the byproducts transported into the mitochondria. Here in the polyamine pathway, they are again further metabolised to yield glutamate and succinic semialdehyde (SUCSAL); this is used to feed the CAC via oxidation to succinic acid directly. In the case of CHIKV and the host, arginine is also utilised for the production of GLUDAC, required for

cysteine biosynthesis.

The host, CHIKV and ZIKV utilise the mitochondrial ORN synthesised glutamate and glutamate 5-semialdehyde (GLU5SA). In the case of DENV, mitochondrial glutamate is phosphorylated to glutamate 5-phosphate and then oxidised to yield GLU5SA. From here, in all optimisation states, GLU5SA is metabolised to 1-Pyrroline-5-carboxylate (1PYR5C) which is then used for the biosynthesis of proline (which is subsequently transported out of the mitochondria). However, in the case of ZIKV, there is a split conversion of 1PYR5C to both proline (via P5CRxm) and glutamate (via P5CDm) with the latter retaining the higher biosynthetic flux (0.22 vs 0.60 mmol glucose^{-1} h^{-1}): a 2.7 fold increase.

Glucose utilisation and metabolic resource allocation highlights different demands of virus and host objectives As described in the previous sections, across all metabolic pathways there is a predicted difference in the utilisation of the metabolic network in the host optimised state, when compared to the virus optimised states. Indeed, there is also a difference in the utilisation of the metabolic network between the different CHIKV-, DENV- and ZIKV-optimised systems. These metabolic rearrangement dissimilarities appear to arise from the differences in the composition of the respective biomass objective functions, for both host and viruses.

For example, DENV has a high requirement for methionine (Fig. 2.4b) which is higher than that of either CHIKV or ZIKV, and consequently is associated with the smallest predicted reaction fluxes responsible for methionine catabolism (METAT). CHIKV has the highest requirement for cysteine of all the viruses (Fig. 2.4a), which is reflected in the predicted fluxes for cysteine biosynthesis (CYSTGL). Additionally, the CHIKV optimal state is the only state that has activation of GACMTRc, the reaction responsible for the conversion of methionine into cysteine synthesis precursors. ZIKV also exhibits similar patterns in the predicted fluxes; large fluxes are predicted for the reactions within the glycine biosynthetic pathway, of which ZIKV has a much higher biomass requirement than other viruses (Fig. 2.4c).

In essence, the virus-optimal flux predictions highlight the rearrangement of metabolic network resources to best suit the current objective that is being optimised for, and this reflects the different compositions of the virus, and the host, biomass objective functions. However, the real power in this technique comes from comparing

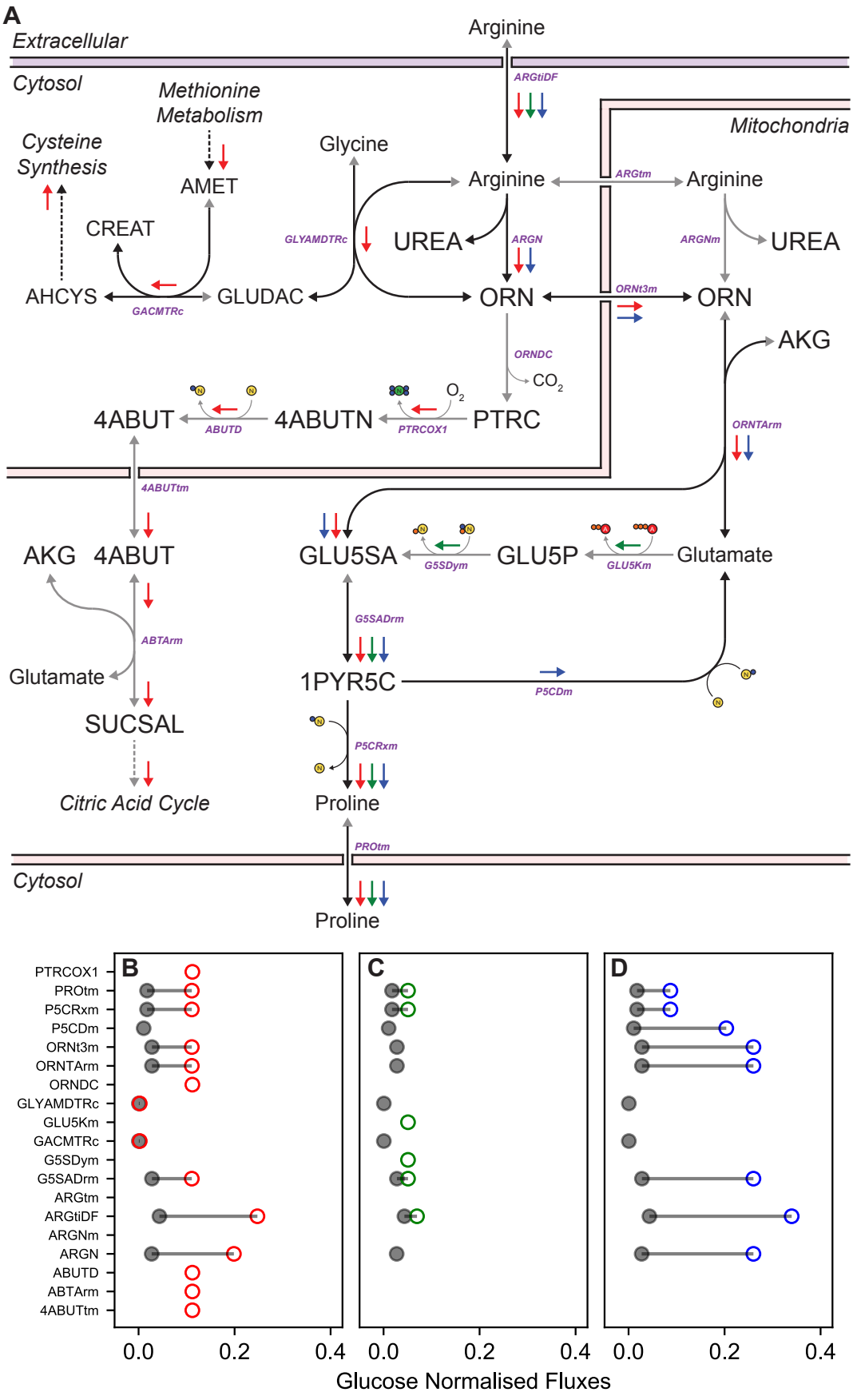

A
Extracellular
Arginine
ARGtiDF
Cytosol
Methionine Metabolism
Mitochondria
Cysteine Synthesis
AMET
Glycine
Arginine
ARGtm
Arginine
CREAT
GLYAMDTRc
UREA
ARGN
ARGNm
UREA
AHCYS
GACMTRc
GLUDAC
ORN
ORNt3m
ORN
ORNDC
CO2
AKG
4ABUT
ABUTD
4ABUTN
O2
PTRCOX1
PTRC
ORNTArm
4ABUTtm
AKG
4ABUT
GLU5SA
G5SDym
GLU5P
GLU5Km
Glutamate
G5SADrm
ABTArm
Glutamate
1PYR5C
P5CDm
SUCSAL
P5CRxm
Citric Acid Cycle
Proline
PROtm
Cytosol
Proline
B
C
D
PTRCOX1
PROtm
P5CRxm
P5CDm
ORNt3m
ORNTArm
ORNDC
GLYAMDTRc
GLU5Km
GACMTRc
G5SDym
G5SADrm
ARGtm
ARGtiDF
ARGNm
ARGN
ABUTD
ABTArm
4ABUTtm
0.0
0.2
0.4
Glucose Normalised Fluxes

Figure 2.13 ***(previous page)*****:** Arginine and Proline biosynthesis metabolic pathway map (**A**) with comparison of predicted host- and virus- optimised flux values (loopless FBA) for (**B**) CHIKV, (**C**) DENV, and (**D**) ZIKV host-virus integrated metabolic models. Arrows indicate reactions, arrowheads indicate potential directionality (two arrowheads indicate a bi-directional reaction). Dashed arrows indicate a connecting set of reactions or pathway where flux has not been explicitly indicated. Host fluxes are indicated through black arrows (➔) with directionality indicated by black arrowhead (grey arrows ➔ and arrow heads indicate no usage of reaction or directionality). Virus optimisations and directionality are indicated by coloured arrows for CHIKV (➔), DENV (➔) and ZIKV (➔). Virus optimised fluxes are compared to host optimised fluxes through normalisation against glucose. Optimised fluxes are indicated through circle markers: Host (●); CHIKV (○); DENV (○); ZIKV (○). Full reaction and metabolite names are given in Appendix B and C, respectively. CHIKV, Chikungunya virus; DENV, Dengue virus; ZIKV, Zika virus.

the viruses against each other, and more importantly against the host-optimal network.

2.3.3 Virus optimal states highlight the differences between virus, but also between viruses and host, in metabolic resource allocation

Comparison of the host and virus optimised systems, in the three iHVMs for CHIKV, DENV and ZIKV has highlighted the differential usage of metabolic resources and the associated reaction pathways. Specifically, glucose is utilised differentially both between the host and the viruses, but also between individual viruses.

All viruses exhibit a higher requirement for nucleotide synthesis, ranging from 13- to 35-times higher flux observed through the *de novo* nucleotide precursor supplying reaction (PRPPS) when virus-optimised compared to host-optimised. This flux is achieved by diversion of glycolytic flux, from glycolysis, to the production of pentose phosphate pathway intermediates for the oxidative (CHIKV and ZIKV) and non-oxidative (DENV) branches. While CHIKV and DENV commit 44-48% of the glycolytic-associated flux to the pentose phosphate pathway, ZIKV fully commits all glucose catabolic flux to the production of the nucleotide biosynthesis precursors.

De novo biosynthesis of amino acid requires significant metabolic resources, many

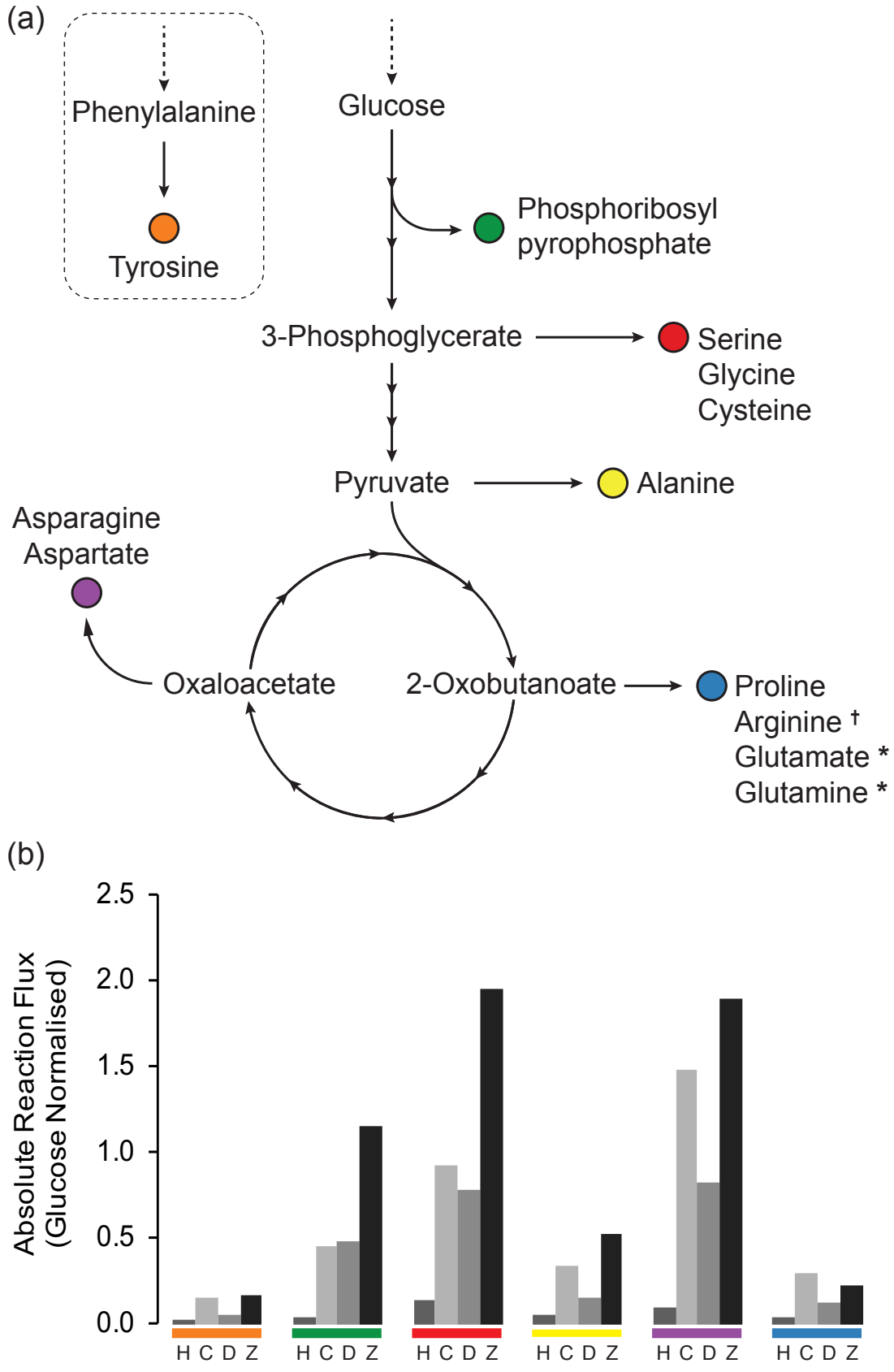

(a)
Phenylalanine
Tyrosine
Glucose
Phosphoribosyl
pyrophosphate
3-Phosphoglycerate
Serine
Glycine
Cysteine
Pyruvate
Alanine
Asparagine
Aspartate
Oxaloacetate
2-Oxobutanoate
Proline
Arginine †
Glutamate *
Glutamine *
(b)
Absolute Reaction Flux
(Glucose Normalised)
2.5
2.0
1.5
1.0
0.5
0.0
H C D Z
H C D Z
H C D Z
H C D Z
H C D Z
H C D Z

Figure 2.14 ***(previous page)*****:** Summary of the production of biomass precursors via *de novo* biosynthetic pathways. **(a)** Cartoon schematic of amino acid biosynthesis in the human macrophage iAB-AMO-1410 metabolic model (Bordbar et al. 2010). Solid arrows depict shortened pathways, dash arrows indicate direct transport from extracellular sources. Amino acids are grouped by their precursor molecule and assigned a colour circle. **(b)** Summary of the biosynthetic flux values, attributed to each precursor as previously identified, for the host (H), Chikungunya virus (C), Dengue virus (D), and Zika virus (Z) optimised metabolic models. Colours are associated with reactions, for summary, as indicated. Orange (PHETHPTOX2), Phenylalanine; green (PRPPS), glucose; red (PGCD), 3-phosphoglycerate; yellow (ALATA_L), pyruvate; purple (ASPTA / ASPTAm), oxaloacetate; blue (P5CRxm), 2-oxobutanoate.
∗ Glutamate and glutamine are involved in too many metabolic reactions, including interconversions of other amino acids, to accurately summarise the biosynthetic fluxes.
† Arginine can be synthesised *de novo* however it is imported from the extracellular environment in this modelling setup.

of which are derived from glucose (Fig. 2.14) apart from Tyrosine which is synthesised directly from Phenylalanine. Across all three viruses there are large increases in amino acid biosynthetic fluxes, as discussed in the previous section. This is evident in the increased fluxes seen throughout the glycolytic and CAC associated reactions. The CAC appears dysfunctional in the DENV- and ZIKV-optimised systems, with an incomplete metabolism of CAC into OAA; under host- and CHIKV-optimisation the CAC runs complete from CIT to OAA via the CAC intermediates. However, in both the host- and ZIKV-optimised systems the CAC is supplemented by anaplerotic metabolism of glutamine into AKG and OAA (glutaminolysis). In the virus-optimised state, cytosolic pyruvate does not directly enter the CAC (via mitochondrial transport) appearing to decouple the CAC from glycolysis.

2.4 DISCUSSION

To simulate the metabolic interactions that occur during viral infections, a novel methodology was created and implemented that represents a virus as a metabolic pseudo-reaction, utilising the virus structural and composition data obtained from bioinformatic databases (Geer et al. 2010). These reactions are comprised of nucleotides (RNA), amino acids and the metabolites associated with the energetic

requirements for the formation of the viruses. Due to a lack of available information regarding the ultrastructure and stoichiometric of the lipid envelopes of CHIKV, DENV and ZIKV, the lipid components of these viruses were omitted from the pseudo-reactions. With available information, the inclusion of a lipid component would improve the overall accuracy of the results. This pseudo-reaction was successfully integrated into a metabolic reconstruction of a human macrophage cell (Bordbar et al. 2010) for three virus species: Chikungunya, Dengue and Zika viruses. These models (MO-CHIKV, MO-DENV and MO-ZIKV, respectively) were analysed using flux balance analysis to predict differences between the metabolic state of an uninfected and infected metabolic network.

Comparison of the biomass compositions of the host and virus-associated pseudo-reactions (Fig. 2.4) highlighted the differential compositions between them. Whilst amino acids are utilised by host and viruses in various ways (Fig. 2.5) it is most notable that nucleotides represent a much more extensive component of virus biomass (Fig. 2.6) than that of the host. This is reflected in multiple comparative predictions of the uninfected and infected models. The requirement for phosphate import exchange is higher than that of the host for all three viruses (Fig. 2.7) which is further demonstrated through the increased flux predicted for all viruses through the pentose phosphate pathway (Fig. 2.9).

Comparison of the usage of the iHVMs in the host- and virus-optimised states highlighted several key areas of host-metabolism that are altered during the simulated infection. Differences in the uptake of metabolites from the *in silico* media were highlighted, showing that virus-optimised systems require more exogenous amino acids (in particular Glutamine) and oxygen than the host. Altered cellular uptake of extracellular metabolites have been observed experimentally in human and mammalian cells with ongoing viral infections, and support (in this case) the idea of differential oxygen (Palomares et al. 2004, Kussow et al. 1995) and glutamine (Chambers et al. 2010, Sanchez et al. 2015, Newsholme et al. 1986, Fontaine et al. 2014, Yu et al. 2011*a*, Carinhas et al. 2017, Goodwin et al. 2015) usage by virus-infected cells.

Comparison of the individual central metabolic, de novo nucleotide and amino acid biosynthesis highlighted the different metabolic strategies of the virus-optimised systems, both compared to the host but also to other viruses. Regardless of the virus being optimised, all virus-optimisations resulted in an increased flux through glycolysis-associated reactions and pathways, such as the pentose phosphate path-

way and serine biosynthetic reactions. This glycolytic metabolic regulation does not solely occur at the initial steps of glycolysis, but occurs throughout the glycolytic pathway. Under the control of CHIKV and DENV, there is an activation of MGSA reactions which convert DHAP to pyruvate. This can indicate that phosphate has become a limiting factor in the metabolic system and thus glyceraldehyde-3-phosphate dehydrogenase cannot maintain a high-enough flux for necessary pyruvate supply(Matsuoka & Shimizu 2013). This highlights predicted virus-specific regulation of multiple metabolic network points, supporting experimental observations in cases of DENV infection (Fontaine et al. 2015, Fischl & Bartenschlager 2011). Under virus control, there is also activation of multiple gluconeogensis associated reactions, and in the case of ZIKV the entire lower part of glycolysis is reversed. This switching of the glycolytic pathway, from glycolysis to gluconeogensis has been observed in multiple viral infections and is attributed to the virus biosynthetic requirements (Deng et al. 2011, Jhuang et al. 2015)

Overall, this analysis successfully develops, describes and compares the metabolic processes of a virus-controlled metabolic network. The differences in metabolic resource allocation can be directly attributed to the biosynthetic needs of the viruses, and appear to increase associated biosynthetic-fluxes far above that of the hosts. The link between the increase in metabolic output and observable disease is still tenuous, and will require further research, however this modelling platform now provides a suitable foundation for the exploration of metabolism-based viral perturbations.

CHAPTER 3

Single-Reaction Perturbations of the Chikungunya, Dengue and Zika virus Integrated Metabolic Models

3.1 INTRODUCTION

Virus infections can be viewed as an interaction between the virus and its hosts metabolic network, causing a rearrangement of metabolic resources to facilitate virus production (Maynard et al. 2010, Sanchez & Lagunoff 2015, Munger et al. 2008). This entanglement between host metabolism and virus production opens up the possibility of perturbing the re-arrangement of host metabolic resources in order to disrupt viral production (Ikeda & Kato 2007, Karupiah & Harris 1995).

Chapter 2 resulted in the creation of three iHVMS for CHIKV, DENV and ZIKV. Subsequent metabolic analysis of these models highlighted the differential usage of reactions and metabolites between host- and virus-optimised systems, and indeed between the CHIKV-, DENV- and ZIKV-optimised systems themselves. To explore the possibility of metabolic-based antiviral therapeutics, these stoichiometric models and their optimisation through FBA can provide an ideal starting point to demonstrate and predict points of importance in the virus-controlled network.

Here, we develop and apply an FBA-based approach to analyse the iHVMs for

Manuscript associated with this chapter is currently under submission.

CHIKV, DENV and ZIKV (MO-CHIKV, MO-DENV and MO-ZIKV respectively). Analysis of these models, in terms of perturbing their metabolic flux distribution when optimised for the virus objective function (VBOF), highlights reactions that can be constrained in order to limit virus production. While other host-virus metabolic modelling studies have focused on using the models as a tool to further understand the interplay between virus infection and host metabolic alterations (Jain & Srivastava 2009, Birch et al. 2012), here we expand upon this and actively search for metabolic antiviral targets. We show that our novel methodology recovers most of the known targets of existing RNA-virus antiviral drugs while highlighting a set of currently unexplored reactions. These reactions are able to limit viral production, either broadly for all three viruses or for a specific virus.

The resulting output from these predictions is a set of reactions combined with suggested inhibitory chemical compounds, identified from the DrugBank (Wishart et al. 2008) database, for CHIKV, DENV and ZIKV. Sensitivity analysis of the antiviral drug targets was conducted through alterations to the host model media assumptions (*in silico* extracellular metabolite alterations) and through evaluation of over 10,000 computationally generated mutant viruses (for CHIKV, DENV and ZIKV).

3.2 METHODS

3.2.1 Integrated Host-Virus Metabolic Model Selection and Creation

Three virus species were selected, as previously defined in §2.2.1, Table 2.1: Chikungunya virus (CHIKV), dengue virus (DENV) and zika virus (ZIKV). VBOFs and integrated host-virus metabolic models were created for each individual virus as described previously in §2.2.2.

3.2.2 Single Reaction Perturbations of Integrated Host-Virus Metabolic Models

Exploration of potential antiviral targets, in the form of constrained reactions, require the definition of new linear optimisation problems that can be applied to the integrated host-virus metabolic models. The previous flux balance analysis optimisation problem (§1.7, Eq. 1.9) is expanded to test two conditions for model

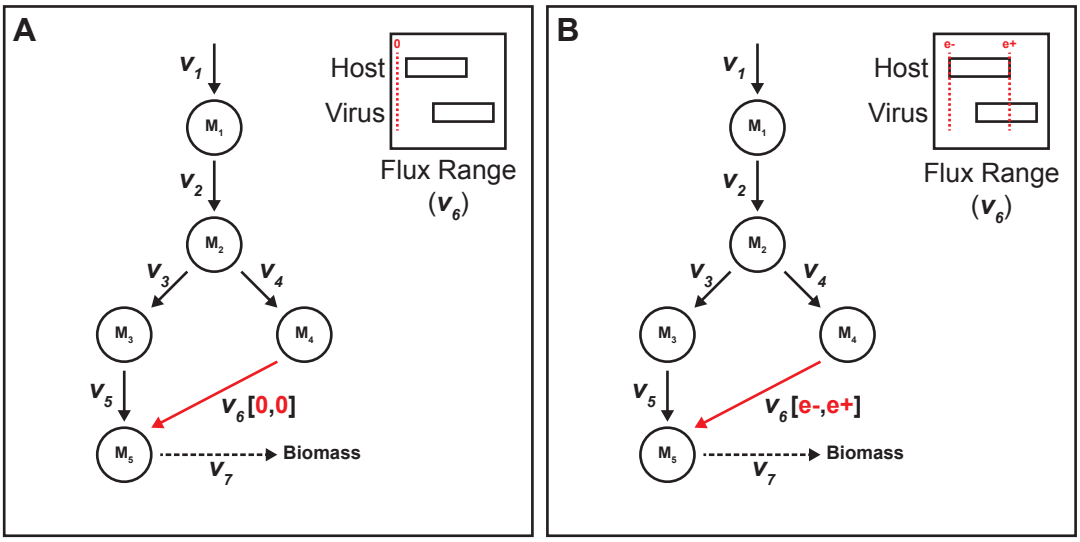


Figure 3.1: Toy model of flux balance analysis optimisation problems for (**A**) single-reaction knockouts and (**B**) single-reaction host-derived enforcements. Toy networks show metabolites (M) as nodes and reactions as edges, with associated reaction flux (V) marked. Flux ranges are shown for the enforced reactions, where the minimum ($e-$) and maximum ($e+$) enforcement bounds are highlighted.

reactions: knockouts and host-derived enforcements. Toy model examples for both single reaction knockouts and single reaction host-derived enforcements are shown in Fig. 3.1.

Single Reaction Knockouts

Knockout analysis considered the effect of systematically setting individual reaction fluxes to zero, and then attempting to maximise the VBOF. This results in a knockout optima for the VBOF (Z_{ko}), which can then be compared against the original VBOF optima (Z_{wt}). This is performed systematically, for a single reaction, over all non-objective function reactions in the metabolic model (Eq. 3.1).

$$\begin{aligned} \text{maximise} \quad & Z_{ko} = \mathbf{c}^T \cdot v \\ \text{subject to} \quad & S \cdot v = 0 \\ & lb < v < ub \\ & 0 < v_i < 0 \end{aligned} \tag{3.1}$$

where indexation i is over all reactions v in the model systematically and the associated lower (lb) and upper (ub) bounds are simultaneously set to zero. This model is then optimised for VBOF, yielding the virus knockout optima (Z_{ko}). A reaction knockout is defined as effective if $Z_{ko} < Z_{wt}$, indicated that knockout of the targeted reaction does result in reduced virus optima. This reduction in virus optima signifies a decrease in the amount of viral biomass that can be produced, which is used as a proxy for decreased virus particle production.

Single Reaction Host-Derived Enforcements

Host-derived enforcements consider the effect of maintaining specific reaction fluxes as they were in a host-optimised state, whilst attempting to optimise the model for a VBOF. The bounds for reactions are systematically set so that the lower and upper flux bounds relate to a specific flux range, e. The range is derived from the corresponding minimum (F^-) and maximum (F^+) flux values, for the individual reaction, obtained from the FVA solution for the host (H) and virus (V) optimal solutions. The range (e) is bounded by minimum (e^-) and maximum (e^+) flux values, which are calculated from conditional arguments (Eq 3.2, 3.3 and 3.4; Fig. 3.1).

$$\begin{aligned} &\text{if} && F_H^+ > F_V^+, F_F^- \geq F_V^- \\ &\text{then} && e^- = F_H^+ - \left(\frac{F_H^+ - F_V^+}{2}\right) \\ & && e^+ = F_H^+ \end{aligned} \tag{3.2}$$

$$\begin{aligned} &\text{if} && F_H^- < F_V^-, F_F^+ \leq F_V^+ \\ &\text{then} && e^- = F_H^- \\ & && e^+ = F_H^- - \left(\frac{F_H^- - F_V^-}{2}\right) \end{aligned} \tag{3.3}$$

$$\begin{aligned} &\text{if} && F_H^- > F_V^-, F_F^+ < F_V^+ \\ &\text{then} && e^- = F_H^- \\ & && e^+ = F_H^+ \end{aligned} \tag{3.4}$$

These calculated flux ranges for each individual reaction are then applied as an additional constraint to the integrated host-virus metabolic model, through alter-

ation of the linear optimisation problem, which is subsequently optimised for the VBOF. Similar to the knockout condition, a new linear optimisation problem is constructed for the single reaction host-derived enforcement (Eq. 3.5).

$$\begin{aligned} \text{maximise} \quad & Z_e = \mathbf{c}^T \cdot v \\ \text{subject to} \quad & S \cdot v = 0 \\ & lb < v < ub \\ & e_i^- < v_i < e_i^+ \end{aligned} \tag{3.5}$$

where indexation i is over all reactions v in the model systematically and the associated lower and upper bounds are set to the corresponding flux minimum and maximum values, as calculated earlier. This model is then optimised for the VBOF, yielding the virus host-derived enforcement optima Z_e. A reaction host-derived enforcement is considered 'effective', in terms of its potential as an antiviral target, if $Z_e < Z_{wt}$.

3.2.3 Generation of alternative virus biomass objective functions and statistical analyses

To evaluate the sensitivity of the results of the single-reaction analysis to possible changes in virus sequences, a mutation-based approach was chosen. This approach allows the consideration of how specific the predicted results of the analysis are to individual viruses (original, wild-type viruses), or could simply have arisen due to the biomass function(s) sharing identical biomass precursors (i.e. amino acids and nucleotides) independent of individual metabolite stoichiometries. A range of VBOFs were generated that were created from mutated forms of the associated virus nucleotide and subsequent protein sequences in order to achieve a compositional gradient away from the known viruses. To this end, two different groups of mutations were implemented: limited nucleotide variations and extensive nucleotide with structural variation.

Limited Nucleotide Variations

To evaluate impact of small deviations from the original VBOFs, variants of the original virus genomes were generated through nucleotide substitution (Fig. 3.2), where the number of nucleotides altered through point mutations were 1, 2, 3, 4,

5 or 10 substitutions. For the subsequent VBOF generation from these variant genomes, the genome and protein copy numbers were kept as in the original (for the respective virus genera).

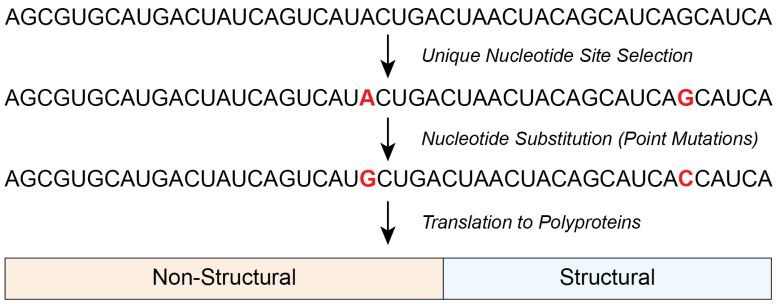


Figure 3.2: Schematic for the point mutation of the virus genomes, for the point mutation sensitivity analysis of host-derived flux enforced reactions. Unique nucleotide sites are selected from the original wild-type genome for a given number of point mutation sites (1, 2, 3, 4, 5 or 10)

To evaluate more variant VBOFs, another 1000 genomes were generated for each virus that were created from the original genome with a random number (between 0 and the total length of the genome) of nucleotide substitutions and using randomly drawn structural and non-structural polyprotein copy numbers per virus particle. This approach is denoted as the "Virus-Like Particle" (VLP) approach, due to the constraints on the length of the genome and the inclusion of the original polyprotein structures.

Extensive Nucleotide with Structural Component Variations

Finally, and in an attempt to generate a set of VBOFs that are far removed from the original ones in terms of compositional gradient, both in terms of genome sequence and the structural and non-structural protein numbers, random VBOFs were directly generated. This is implemented by drawing 1000 sets of individual stoichiometries of biomass components from a uniform distribution on [a, b], where a and b are (1) ±99% of the original stoichiometric coefficients of a given virus, or (2) are ±99% of the average of all original stoichiometric coefficients of a given virus. These approaches to generating variant virus genomes yield a set of sequences (and associated VBOFs) that are increasingly removed from the original virus VBOFs.

For each randomised VBOF created, the host-derived flux enforcement analysis

is repeated (with a recalculation of the bounds used for the individual enforcement), and the reactions that perturb virus optima the most when constrained are identified. This whole analysis resulted in 8000 randomised VBOFs and FBA simulations, the results of which are summarised as percentage impact of individual host reactions on virus optima for different sets of VBOFs.

To compare results of flux enforcement analysis to that obtained from using randomised biomass functions, a one-way ANOVA and Tukey's honest significance test were used for each individual virus (which includes the point mutations) against the randomised virus groups (VLP and random).

3.2.4 Measuring impact of alternative host model assumptions on single-reaction analyses

The host macrophage model uses specific metabolite uptake fluxes, which are mostly based on experimental observations (Bordbar et al. 2010), but which can directly influence FBA-based results. To evaluate the potential impact of altered model uptake bounds, virus optimisation was reanalysed using alternative metabolite uptake fluxes in the host model in a systematic fashion. First, metabolites are identified that are supplied (via exchange reactions) to the metabolic model with non-arbitrary lower bounds (lb), where $0 > lb > -\inf$, as previously defined in §2.5 Table 2.8. Each of the identified reactions are then systematically constrained, such that the lb is reduced from the original model values (Bordbar et al. 2010) (in steps of 10%) until the $lb = 0$, effectively knocking-out the respective exchange reaction. For each altered (additionally constrained) model, the host-derived enforcement analysis is repeated, with re-calculation of the viable host reaction bounds and optimisation of VBOF, done for each of the three viruses (CHIKV, DENV, ZIKV).

3.3 RESULTS

Results presented encompass predictions of potential antiviral targets through two reaction conditions: knockouts and host-derived enforcements. All analyses were conducted for CHIKV, DENV and ZIKV in their respective iHVMs and are presented as "broad targets" (all viruses affected) and "specific targets" (one or two viruses are affected). Effectiveness is defined as when the target reaction results

in a virus optima that is less than the unconstrained virus optima.

As highlighted in chapter 2 the host-optimal and virus-optimal flux distributions can differ, therefore we hypothesis that the iHVM can be constrained in a way to limit viral production. Here, such constraints are applied in two different ways: reaction knockouts (Eq. 3.1) and host-derived enforcements (Eq 3.5).

Effective targets, able to reduce the virus optima (as previously defined), are cross-checked against the DrugBank database (Wishart et al. 2008) for compounds that are verified or predicted to inhibit the aforementioned reaction. This cross-reference check is completed using the reaction-associated enzyme commission (EC) number, which links enzymes to specific reactions (Schomburg 2004). These compounds can then be utilised for experimental validation.

3.3.1 Single Reaction Knockouts Highlight Metabolic Overlap of Host and Virus Optimal States

The single reaction knockout analysis predicts that there are individual reactions that are able to completely inhibit virus biomass optimisation; essentially, the virus is unable utilise some biomass precursors for it's own production, under the affects of the additional constraints in place from the single reaction knockout analysis.
.

In total, there were 41 reactions that were predicted to be effective against all three viruses: CHIKV, DENV and ZIKV. These reactions, including numbers of potential drug candidates for antiviral therapeutic use, are listed in Table 3.1 (full results available in Appendix D). All targets presented result in a reduced virus optima (in all cases) of zero, effectively meaning that no viral biomass is produced under that particular reaction constraint condition. The one exception to this is an additional reaction that was predicted to be effective against ZIKV, and reducing the associated optima to 92% of the original, wild-type optima: ATPS4m, a mitochondrial reaction that is involved in oxidative phosphorylation.

However, whilst these predicted reactions provide potential antiviral targets against CHIKV, DENV and ZIKV, they also reduce the viability of the host metabolic network. Under the additional constraints imposed by the single reaction knock-out analysis, any reaction that reduces the virus optima to zero also inadvertently reduces the host optima to zero. Essentially, there does not exist a single reaction

knockout that can be effective at reducing virus optima, whilst simultaneously permitting the host to successfully optimise and utilise the metabolic network.

Table 3.1: Summary of 41 reactions that are predicted to have an antiviral effect whilst under a single reaction knockout condition. Number of drug candidates are identified from the DrugBank (Wishart et al. 2008) database if matched with an UniProt Identifier (derived from the EC Number). The knockout targets listed are predicted to be effective against all three viruses: Chikungunya, Dengue and Zika. In all cases the virus knockout optima was reduced to 0. EC Number, Enzyme Commission Number (Schomburg 2004).

Reaction Information			**Number of Drug Candidates**
Model Reaction ID	Model Subsystem	EC Number	
ADSL1	Nucleotides	4.3.2.2	0
ADSL2	IMP Biosynthesis	4.3.2.2	0
ADSS	Nucleotides	6.3.4.4	7
AHC	Methionine Metabolism	3.3.1.1	4
AICART	IMP Biosynthesis	2.1.2.3	9
AIRCr	IMP Biosynthesis	4.1.1.21	1
ALATA_L	Glutamate metabolism	2.6.1.2	4
ASNS1	Alanine and Aspartate Metabolism	6.3.5.4	4
ASPCTr	Pyrimidine Biosynthesis	2.1.3.2	2
CBPS	Pyrimidine Biosynthesis	6.3.5.5	2
CTPS1	Nucleotides	6.3.4.2	1
CYOR-u10m	Oxidative Phosphorylation	1.10.2.2	9
CYSTGL	Cysteine Metabolism	4.4.1.1	5
CYSTS	Methionine Metabolism	4.2.1.22	4
DHORD9	Pyrimidine Biosynthesis		

Continued on next page

Table 3.1 – continued from previous page

Model Reaction ID	Model Subsystem	EC Number	Number of Drug Candidates
DHORTS	Pyrimidine Biosynthesis	3.5.2.3	2
G5SADrm	Arginine and Proline Metabolism		
GARFT	IMP Biosynthesis	2.1.2.2	3
GHMT2r	Glycine, Serine, and Threonine Metabolism	2.1.2.1	7
GK1	Nucleotides	2.7.4.8	1
GLUPRT	IMP Biosynthesis	2.4.2.14	2
GMPS2	Nucleotides	6.3.5.2	1
IMPC	IMP Biosynthesis	3.5.4.10	9
IMPD	Nucleotides	1.1.1.205	7
METAT	Methionine Metabolism	2.5.1.6	3
O2t	Transport, Extracellular		
OMPDC	Pyrimidine Biosynthesis	4.1.1.23	1
ORPT	Pyrimidine Biosynthesis	2.4.2.10	1
P5CRxm	Arginine and Proline Metabolism	1.5.1.2	2
PGCD	Glycine, Serine, and Threonine Metabolism	1.1.1.95	1
PHETHPTOX2	Tyr, Phe, Trp Biosynthesis	1.14.16.1	9
PRAGSr	IMP Biosynthesis	6.3.4.13	3
PRAIS	IMP Biosynthesis	6.3.3.1	3

Continued on next page

Table 3.1 – continued from previous page

Model Reaction ID	Model Subsystem	EC Number	Number of Drug Candidates
PRASCS	IMP Biosynthesis	6.3.2.6	1
PRFGS	IMP Biosynthesis	6.3.5.3	1
PROtm	Transport, Mitochondrial		
PRPPS	Pentose Phosphate Pathway	2.7.6.1	0
PSERT	Glycine, Serine, and Threonine Metabolism	2.6.1.52	2
PSP_L	Glycine, Serine, and Threonine Metabolism	3.1.3.3	1
RPI	Pentose Phosphate Pathway	5.3.1.6	1
THBPT4ACAMDASE	Tetrahydrobiopterin	4.2.1.96	1

Overall, this knockout-driven analysis highlights that, whilst a number of reactions can be fully inhibited and completely stop the virus from being able to produce biomass within the constrained metabolic network, there are no reactions that are capable of reducing the virus optima whilst simultaneously maintaining any aspect of host viability. In order to explore potential antiviral-yielding metabolic constraints, that maintain host viability (i.e. the host optima is non-zero), a novel methodology was developed that requires the enforcement of a host optimal state on the metabolic network.

3.3.2 Host-Derived Enforcements Provide Potential Antiviral Effect

Following from the knockout results, the host-derived enforcement analysis was developed. This analysis ensures that host viability is entirely preserved (i.e. the host optima does not decrease from the original unconstrained host optima), while virus optima is reduced below wild-type levels. This is achieved through maintenance of host metabolic resource allocation through reaction flux manipulation, specifically alteration of the lower and upper reaction bounds (Eq 3.5).

The enforcement analysis predicted several potential antiviral targets, many of which were also identified in the knockout analysis. These antiviral targets span a variety of subsystems, and are shown to have an effect on all three viruses to some degree (Fig. 3.3). There was also another set of antiviral targets predicted, that are virus-specific in terms of their effects; not all virus optima are reduced when additional host-derived constraints are applied to the reaction (Fig. 3.5).

There are a subset of reactions that were not predicted as antiviral targets in host-enforcement analysis, when they previously had been highlighted in the knockout analysis. This suggests that some reactions cannot be constrained in such a way to simultaneously be an effective antiviral whilst negating damage to the host. Whilst these identified reactions are potential antiviral targets, in the sense that altering their fluxes can limit virus production within the host, they do not contribute to the set of host-viable antiviral targets, as discussed above.

A set of these reactions overlap with known antiviral drug targets, identified through literature analysis. There are currently 10 antivirals specific to RNA viruses which target a total of 5 unique metabolic enzymes (Table 3.2). Of these 5 drug targets (and the associated drugs) one has been experimentally verified to be effective against CHIKV (Inositol-5'-monophosphate dehydrogenase; IMPD)

(Khan et al. 2011) and another against DENV (dihydroorotate dehydrogenase; DHORD9) (Wang et al. 2011). While the other three targets have been verified to be effective against some RNA viruses(Leyssen et al. 2008), they have yet to be tested against CHIKV, DENV and ZIKV.

Table 3.2: List of antiviral compounds identified, which interact with host metabolic reactions that affect RNA virus production (Littler & Oberg 2005, Leyssen et al. 2008). CTP Synthase is encoded for by two genes, CTPS1 and CTPS2 (Sahoo et al. 2012), and both are affected by the associated antiviral compounds (only one gene is shown). All antiviral compounds, and their associated target reactions, were identified from literature search. EC; Enzyme Commission (Schomburg 2004)

.

Compound	**Target Reaction**	**Model Reaction ID**	**EC Number**
Cyclopentylcytosine Cyclopentenylcytosine DD264	Cytidine triphosphate synthetase	CTPS1	6.3.4.2
NITD-982 A3	Dihydroorotate dehydrogenase	DHORD9	1.3.5.2
Ribavirin VX-497	Inosine 5-monophosphate dehydrogenase	IMPD	1.1.1.205
Pyrazofurin 6-azauridine C-c3Ado	Orotidine 5-phosphate decarboxylase	OMPDC	4.1.1.23
neplanocin A 3-deazaneplanocin A aristeromycin	S-adenosylhomocysteine hydrolase	AHC	3.3.1.1

These enforcement reactions are not universal in the viruses that they effect; they can be split into the broad (Fig. 3.3) and specific (Fig. 3.5) target results.

Reactions are identified as being effective against all three viruses under the host-derived enforcement condition (Fig 3.3), with more instances of greater ZIKV optima reduction that than of DENV, which both are more effectively perturbed than CHIKV (Fig 3.3A), due to a greater number of reactions able to reduce the virus optima to below 80% for DENV and ZIKV than CHIKV. Varying effectiveness, in terms of the ability of a reaction to be targeted and reduce virus optima, varies across viruses and subsystems. Of all 46 reactions predicted to be effective broad antiviral targets, only one reaction is able to reduce the optima of all three viruses to below 60%: guanylate kinase (GMP:ATP); GK1. This phosphortransferase catalyses the following reaction:

$$\mathrm{ATP} + \mathrm{GMP} \rightleftharpoons \mathrm{ADP} + \mathrm{GDP}$$

and is a critical step in the synthesis of guanosine containing nucleotides. A selection of reactions, in addition to GK1, that form the main branched pathways of *de novo* nucleotide biosynthesis are shown in Fig. 3.4. Of all identified broad-targeting reactions, the majority (24) fall within these pathways.

From this analysis, 13 subsystems are highlighted as having reactions that can be constrained in such a way to act as potential antiviral targets (Fig 3.3B). The subsystem with the most effective reaction targets is the IMP biosynthesis subsystem, responsible for the metabolism of inosine nucleosides to synthesise adenosine and guanine ribonucleotide bases. In fact, the top three subsystems (including the aforementioned) relate to *de novo* nucleotide biosynthesis, with the second and third most effective subsystems relating to general nucleotide and pyrimidine biosynthesis. Note that whilst reactions are uniquely assigned to subsystems (they cannot be assigned to more than one), many subsystems are directly adjacent to one another in terms of reaction progression (i.e. the "Nucleotides" subsystem feeds and is fed directly by the "IMP Biosynthesis" and "Pyrimidine Biosynthesis" pathways due to the metabolic network topology). In general, nucleotide-associated metabolism represents 4 of the 13 identified subsystems, with the remaining being associated with amino acid metabolism (3/13), co-factor metabolism (2/13), and other miscellaneous transport and side-reactions.

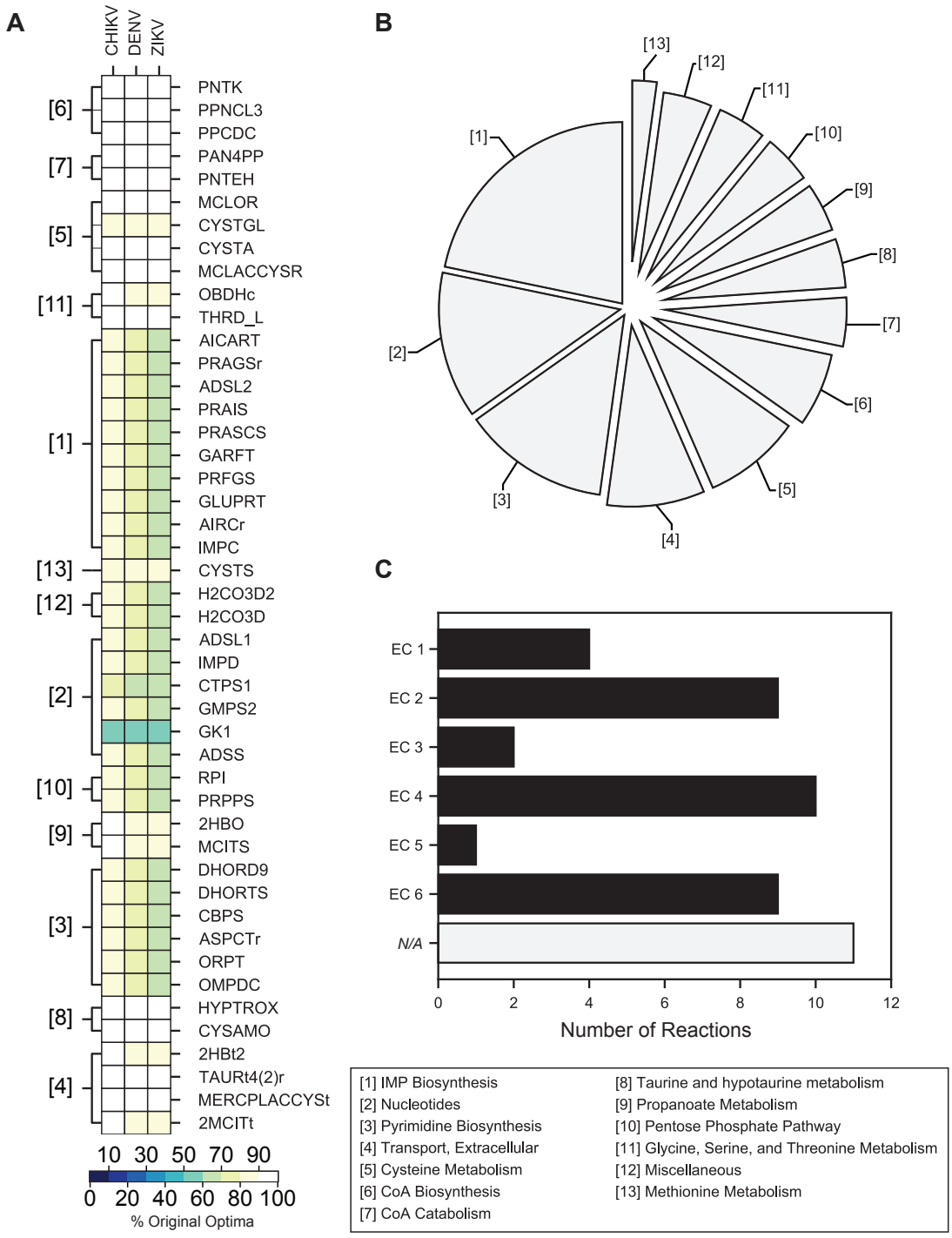


Figure 3.3: Summary of the broad targets (all viruses affected) predicted by single-reaction host-derived enforcement analysis. **(A)** Heat map of the percentage original unconstrained virus optima, for single reactions, under the host-derived enforcement condition and grouped numerically by subsystem (key in bottom right). **(B)** Proportion breakdown of the number of reactions identified for each numerically labelled subsystem.
(continued on next page...)

Figure 3.3 ***(previous page)*****: (C)** Number of reactions that are identified as being effective under host-derived enforcement conditions, and the associated enzyme commission (EC) classification assigned to them (*N/A* denotes no associated EC record for a given reaction). Full list of reaction names can be found in Table 3.3. CHIKV, Chikungunya Virus; DENV, Dengue Virus; ZIKV, Zika virus.

The majority of reactions identified do not have an associated EC number, which identifies the class / type of enzyme that is associated with the reaction (Fig 3.3 C), denoted as *N/A*. Of the rest of the reaction targets identified, the majority belong to one of three enzyme classes: transferases (EC2); lyases (EC4); and ligases (EC6). These three reaction classes are largely responsible for the transformation, catabolism and anabolism (respectively) of metabolic compounds. The majority of these ECs and associated reactions, as discussed, occur in the nucleotide-containing pathways.

Another set of reactions predicted, where those that specifically effect one or two viruses, and as such are known as ”specific targets”. From the host-derived enforcement analysis a total of 58 reactions were predicted to be in this set (Fig 3.5 A). The majority of the reactions predicted reduce virus optima, of one of the three viruses, to 90-100% of the original wild-type virus optima (i.e. a reduction of ¡ 10%). Whilst there are reactions predicted to be effective against all three viruses, the breakdown is not evenly weighted between them; there are 17, 4 and 53 CHIKV, DENV and ZIKV specific targets (respectively). Some of these targets are not virus exclusive, and are also identified in another viruses effective set (Fig 3.5 B). CHIKV and ZIKV share the largest overlap (12 reactions effective against them both) and ZIKV has the highest number of ZIKV-exclusively effective reactions (37). Observation of the reaction subsystem breakdown (Fig 3.5 C) highlights the differential effective subsystem distribution for the three viruses. The most effective subsystems against CHIKV, DENV and ZIKV (respectively) were: valine, leucine and isoleucine metabolism; tryptophan metabolism; extracellular transport. Analysis of the EC numbers of the reactions highlight a different distribution of the largest number of effective reactions in terms of enzyme classes: EC1 oxidoreductases (CHIKV); EC2 transferases (ZIKV). In the case of DENV the distribution is spread evenly (due to the low number of DENV-specific effective reactions) across the EC1, EC2 and EC4 classes.

For both broad- and specific-reaction targets (Tables 3.3 and 3.4), a set of potential drug candidates are identified from the DrugBank (Wishart et al. 2008) compound

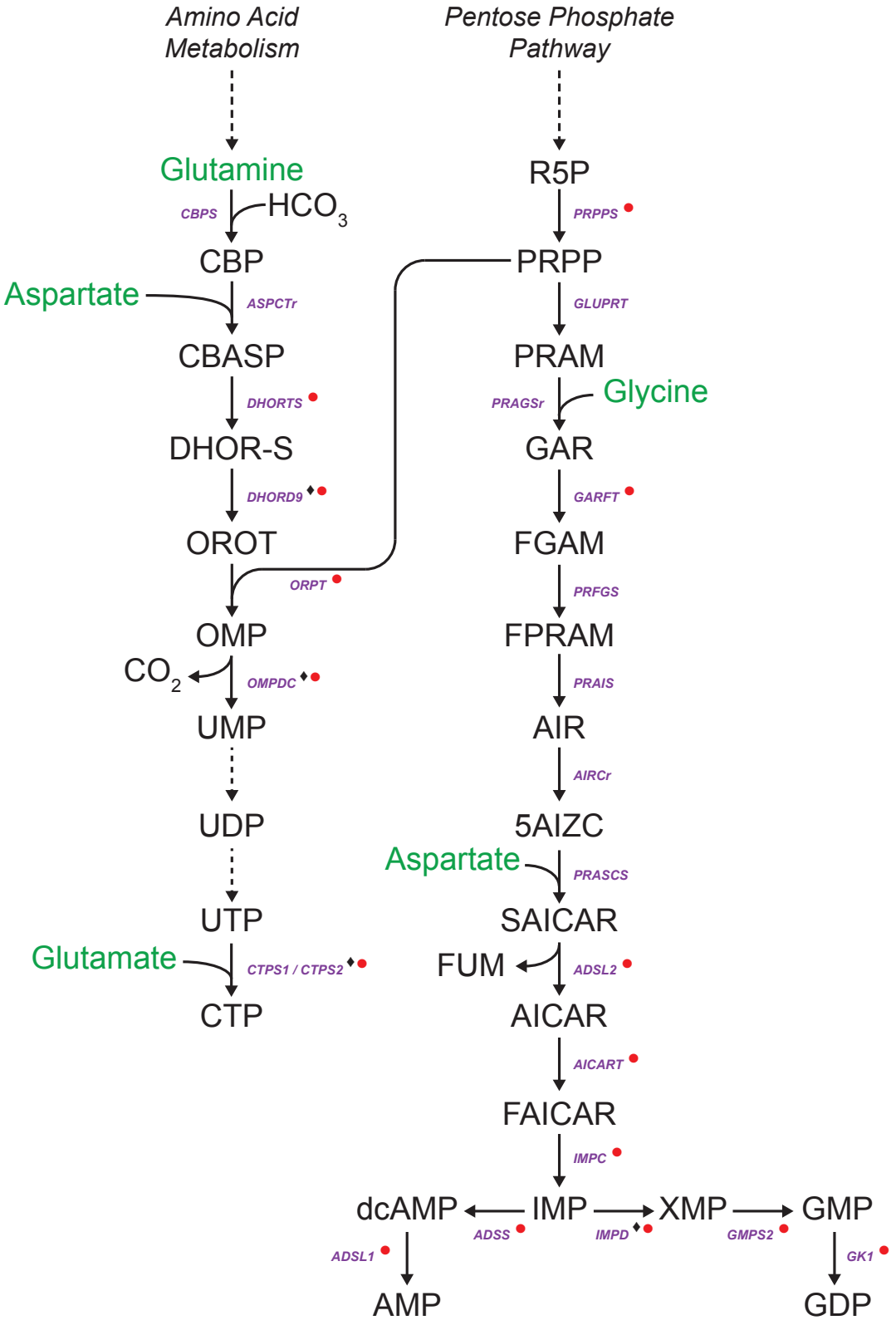


Figure 3.4: Single-reaction host-derived enforcement targets predicted as being effective against Chikungunya (CHIKV), Dengue (DENV) and Zika (ZIKV) viruses. Existing antiviral targets (◆) and reactions with drugs predicted from this analysis (●) are shown. Full reaction names are given in Appendix B.

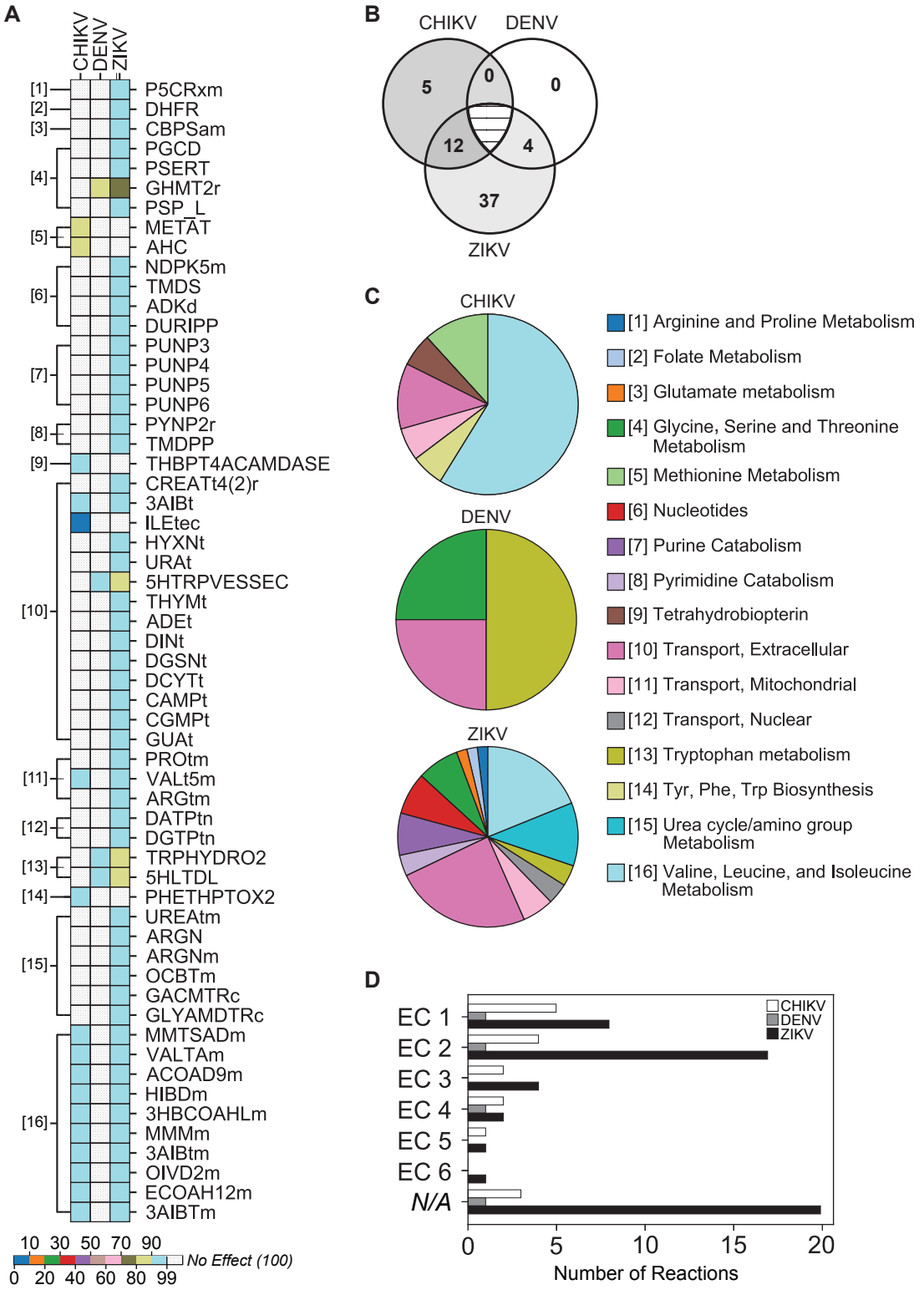


Figure 3.5: Summary of the specific targets (one or two viruses effected) predicted by single-reaction host derived enforcement analysis. **(A)** Heat map of the percentage original unconstrained virus optima, for single reactions, under the host-derived enforcement condition and grouped numerically by subsystem ...
(continued on next page...)

Figure 3.5 ***(previous page)*****:** ... (key in middle right). **(B)** Venn diagram showing the overlap between specific reaction targets, in terms of the viruses they effect. **(D)** Proportion breakdown of the number of reactions that are identified specifically for each virus, subdivided for each colour-coded subsystem. **D** Number of reactions that are identified as being effective under host-derived enforcement conditions, and the associated enzyme commission (EC) classification assigned to them (*N/A* denotes no associated EC record for a given reaction). Full list of reaction names can be found in Table 3.4. CHIKV, Chikungunya Virus; DENV, Dengue Virus; ZIKV, Zika virus.

database, indicated by a "Yes" in the respective table. A full list of identified chemical compounds, identified as effective drugs for the associated reaction target, are provided in Appendix D.

Table 3.3: Reactions predicted to have an antiviral effect whilst under a single reaction host-derived enforcement condition. Drug candidates are listed from the DrugBank (Wishart et al. 2008) database if matched with an UniProt Identifier (derived from the EC Number). Enforcement targets are predicted to be effective against Chikungunya (CHIKV), Dengue (DENV) and Zika (ZIKV) viruses. Full results available in Appendix D. EC Number, Enzyme Commission Number (Schomburg 2004).

Reaction Information			**% Original Optima**			**Drugs Identified**
Model Reaction ID	Model Subsystem	EC Number	CHIKV	DENV	ZIKV	
2HBO	Propanoate Metabolism	1.1.1.27	98	88	89	Yes
2HBt2	Transport, Extracellular		98	88	89	No
2MCITt	Transport, Extracellular		98	88	89	No
ADSL1	Nucleotides	4.3.2.2	89	71	69	No
ADSL2	IMP Biosynthesis	4.3.2.2	86	71	67	No
ADSS	Nucleotides	6.3.4.4	89	71	69	Yes
AICART	IMP Biosynthesis	2.1.2.3	86	71	67	Yes
AIRCr	IMP Biosynthesis	4.1.1.21	86	71	67	Yes
ASPCTr	Pyrimidine Biosynthesis	2.1.3.2	85	70	67	Yes
CBPS	Pyrimidine Biosynthesis	6.3.5.5	85	70	67	Yes
CTPS1	Nucleotides	6.3.4.2	74	65	61	Yes
CYSAMO	Taurine and hypotaurine metabolism		93	93	92	No
CYSTA	Cysteine Metabolism	2.6.1.3	93	93	92	Yes
CYSTGL	Cysteine Metabolism	4.4.1.1	85	81	85	Yes
CYSTS	Methionine Metabolism	4.2.1.22	85	81	85	Yes
DHORD9	Pyrimidine Biosynthesis		85	70	67	No

Continued on next page

Table 3.3 – continued from previous page

Model Reaction ID	Model Subsystem	EC Number	CHIKV	DENV	ZIKV	Drugs Identified
DHORTS	Pyrimidine Biosynthesis	3.5.2.3	85	70	67	Yes
GARFT	IMP Biosynthesis	2.1.2.2	86	71	67	Yes
GK1	Nucleotides	2.7.4.8	50	50	50	Yes
GLUPRT	IMP Biosynthesis	2.4.2.14	86	71	67	Yes
GMPS2	Nucleotides	6.3.5.2	83	71	66	Yes
H2CO3D	Miscellaneous	4.2.1.1	85	70	67	Yes
H2CO3D2	Miscellaneous		85	70	67	No
HYPTROX	Taurine and hypotaurine metabolism		93	93	92	No
IMPC	IMP Biosynthesis	3.5.4.10	86	71	67	Yes
IMPD	Nucleotides	1.1.1.205	83	71	66	Yes
MCITS	Propanoate Metabolism	4.1.3.31	98	88	89	No
MCLACCYSR	Cysteine Metabolism		93	93	92	No
MCLOR	Cysteine Metabolism	1.1.1.27	93	93	92	Yes
MERCPLACCYSt	Transport, Extracellular		93	93	92	No
OBDHc	Glycine, Serine, and Threonine Metabolism	1.2.7.2	98	88	89	No
OMPDC	Pyrimidine Biosynthesis	4.1.1.23	85	70	67	Yes
ORPT	Pyrimidine Biosynthesis	2.4.2.10	85	70	67	Yes
PAN4PP	CoA Catabolism		93	93	92	No
PNTEH	CoA Catabolism		93	93	92	No

Continued on next page

Table 3.3 – continued from previous page

Model Reaction ID	Model Subsystem	EC Number	CHIKV	DENV	ZIKV	Drugs Identified
PNTK	CoA Biosynthesis	2.7.1.33	93	93	92	No
PPCDC	CoA Biosynthesis	4.1.1.36	93	93	92	Yes
PPNCL3	CoA Biosynthesis	6.3.2.5	93	93	92	No
PRAGSr	IMP Biosynthesis	6.3.4.13	86	71	67	Yes
PRAIS	IMP Biosynthesis	6.3.3.1	86	71	67	Yes
PRASCS	IMP Biosynthesis	6.3.2.6	86	71	67	Yes
PRFGS	IMP Biosynthesis	6.3.5.3	86	71	67	Yes
PRPPS	Pentose Phosphate Pathway	2.7.6.1	85	71	67	No
RPI	Pentose Phosphate Pathway	5.3.1.6	85	71	67	Yes
TAURt4(2)r	Transport, Extracellular		93	93	92	No
THRD_L	Glycine, Serine, and Threonine Metabolism	4.3.1.19	95	91	91	Yes
sink_Tyr-ggn			0	0	0	No

Table 3.4: Reactions predicted to have an antiviral effect whilst under a single reaction host-derived enforcement condition, but only against Chikungunya (CHIKV), Dengue (DENV) and Zika (ZIKV) viruses in a specific manner (i.e. Chikungunya only). Drug candidates are listed from the DrugBank (Wishart et al. 2008) database if matched with an UniProt Identifier (derived from the EC Number). Full results available in D. EC Number, Enzyme Commission Number (Schomburg 2004).

Reaction Information			**% Original Optima**			**Drugs Identified**
Model Reaction ID	Model Subsystem	EC Number	CHIKV	DENV	ZIKV	
3AIBTm	Valine, Leucine, and Isoleucine Metabolism	2.6.1.22	97	100	97	Yes
3AIBt	Transport, Extracellular		97	100	97	No
3AIBtm	Valine, Leucine, and Isoleucine Metabolism	2.6.1.22	97	100	97	Yes
3HBCOAHLm	Valine, Leucine, and Isoleucine Metabolism	3.1.2.4	97	100	97	Yes
5HLTDL	Tryptophan metabolism	4.1.1.28	100	91	84	Yes
5HTRPVESSEC	Transport, Extracellular		100	91	84	No
ACOAD9m	Valine, Leucine, and Isoleucine Metabolism	1.3.99.12	97	100	97	No
ADEt	Transport, Extracellular		100	100	96	No
ADKd	Nucleotides	2.7.4.3	100	100	98	Yes
AHC	Methionine Metabolism	3.3.1.1	85	100	100	Yes
ARGN	Urea cycle/amino group metabolism	3.5.3.1	100	100	97	Yes
ARGNm	Urea cycle/amino group metabolism	3.5.3.1	100	100	97	Yes

Continued on next page

Table 3.4 – continued from previous page

Model Reaction ID	Model Subsystem	EC Number	CHIKV	DENV	ZIKV	Drugs Identified
ARGtm	Transport, Mitochondrial		100	100	97	No
CAMPt	Transport, Extracellular		100	100	96	No
CBPSam	Glutamate metabolism	6.3.4.16	100	100	96	Yes
CGMPt	Transport, Extracellular		100	100	97	No
CREATt4(2)r	Transport, Extracellular		100	100	97	No
DATPtn	Transport, Nuclear		100	100	98	No
DCYTt	Transport, Extracellular		100	100	96	No
DGSNt	Transport, Extracellular		100	100	96	No
DGTPtn	Transport, Nuclear		100	100	96	No
DHFR	Folate Metabolism	1.5.1.3	100	100	99	Yes
DINt	Transport, Extracellular		100	100	97	No
DURIPP	Nucleotides		100	100	99	No
ECOAH12m	Valine, Leucine, and Isoleucine Metabolism	4.2.1.17	97	100	97	Yes
GACMTRc	Urea cycle/amino group metabolism	2.1.1.2	100	100	97	Yes
GHMT2r	Glycine, Serine, and Threonine Metabolism	2.1.2.1	100	87	79	Yes
GLYAMDTRc	Urea cycle/amino group metabolism	2.1.4.1	100	100	97	Yes
GUAt	Transport, Extracellular		100	100	96	No

Continued on next page

Table 3.4 – continued from previous page

Model Reaction ID	Model Subsystem	EC Number	CHIKV	DENV	ZIKV	Drugs Identified
HIBDm	Valine, Leucine, and Isoleucine Metabolism	1.1.1.31	97	100	97	Yes
HYXNt	Transport, Extracellular		100	100	96	No
ILEtec	Transport, Extracellular		0	100	100	No
METAT	Methionine Metabolism	2.5.1.6	85	100	100	Yes
MMMm	Valine, Leucine, and Isoleucine Metabolism	5.4.99.2	97	100	97	Yes
MMTSADm	Valine, Leucine, and Isoleucine Metabolism	1.2.1.18	97	100	97	Yes
NDPK5m	Nucleotides	2.7.4.6	100	100	97	Yes
OCBTm	Urea cycle/amino group metabolism	2.1.3.3	100	100	96	Yes
OIVD2m	Valine, Leucine, and Isoleucine Metabolism	1.2.1.25	97	100	97	No
P5CRxm	Arginine and Proline Metabolism	1.5.1.2	100	100	95	Yes
PGCD	Glycine, Serine, and Threonine Metabolism	1.1.1.95	100	100	95	Yes
PHETHPTOX2	Tyr, Phe, Trp Biosynthesis	1.14.16.1	94	100	100	Yes
PROtm	Transport, Mitochondrial		100	100	95	No
PSERT	Glycine, Serine, and Threonine Metabolism	2.6.1.52	100	100	95	Yes

Continued on next page

Table 3.4 – continued from previous page

Model Reaction ID	Model Subsystem	EC Number	CHIKV	DENV	ZIKV	Drugs Identified
PSP_L	Glycine, Serine, and Threonine Metabolism	3.1.3.3	100	100	95	Yes
PUNP3	Purine Catabolism	2.4.2.1	100	100	97	Yes
PUNP4	Purine Catabolism	2.4.2.1	100	100	96	Yes
PUNP5	Purine Catabolism	2.4.2.1	100	100	97	Yes
PUNP6	Purine Catabolism	2.4.2.1	100	100	97	Yes
PYNP2r	Pyrimidine Catabolism	2.4.2.2	100	100	96	No
THBPT4ACAMDASE	Tetrahydrobiopterin	4.2.1.96	94	100	100	Yes
THYMt	Transport, Extracellular		100	100	99	No
TMDPP	Pyrimidine Catabolism	2.4.2.4	100	100	99	Yes
TMDS	Nucleotides	2.1.1.45	100	100	99	Yes
TRPHYDRO2	Tryptophan metabolism	1.14.16.4	100	91	84	Yes
URAt	Transport, Extracellular		100	100	96	No
UREAtm	Urea cycle/amino group metabolism		100	100	97	No
VALTAm	Valine, Leucine, and Isoleucine Metabolism	2.6.1.42	97	100	97	Yes
VALt5m	Transport, Mitochondrial		97	100	97	No

3.3.3 Effect of virus mutations and model alterations on predicted antiviral targets

The results presented so far in this chapter have supported the idea of using the iHVMs for identifying host-based antiviral targets. However, any analysis based on stoichiometric metabolic flux optimisation, as done so here, can be dependent on details of model implementation and assumptions regarding the biomass composition of the modelled organism (Schuetz et al. 2012).

For example, whilst the host metabolic model and the biomass function that it incorporates are verified against experimental data (Thiele & Palsson 2010) the model still assumes a specific media composition and uptake fluxes. Variation in the media and associated exchange fluxes, responsible for supplying the network with the necessary metabolic resources, may alter the antiviral effect that is predicted. In the same vein, alterations of the organisms biomass composition (in this case, alterations of the virus biomass) may result in altered effectiveness of predicted antiviral targets. This can be likened to the effect of virus mutation on subverting current antiviral theraputics, such as those seen in cases of HIV resistance (Littler & Oberg 2005, Perelson & Ribeiro 2013). Therefore, being able to preemptively model such situations, and identify antiviral targets that still remain viable (i.e. able to reduce the virus optima) would potentially prove useful in selecting appropriate targets for experimental studies.

To test the robustness of the predictions in this chapter, the effects of *in silico* media alterations (host-centred) and virus genome mutations (virus-centred) are conducted.

Impact of altered metabolic model *in Silico* media constraints

To test if the host-derived reaction predictions are robust against key assumptions in the model, the effects of variations in the media composition are analysed for the host model in the context of the antiviral target prediction. For each virus (CHIKV, DENV and ZIKV) the host-derived enforcements were reanalysed for 1000 alternative media uptake fluxes for 20 different media metabolites (Table 3.5).

For all viruses, altering the media can affect the perturbation achieved by the host-derived flux enforced reaction. However, for many of the reactions, the media

Table 3.5: *In silico* media components for the human macrophage metabolic model (Bordbar et al. 2010).

In silico Media Component	Model Metabolite ID
Arginine	arg_L_e
Butyrate	but_e
Glucose	glc_e
Glutamine	gln_L_e
Bicarbonate	hco3_e
Histidine	his_L_e
Isoleucine	ile_L_e
Leucine	leu_L_e
Lysine	lys_L_e
Methionine	met_L_e
Oxygen	o2_e
Steric Acid	ocdca_e
Oleic Acid	ocdcea_e
Phenylalanine	phe_L_e
Phosphate	pi_e
Pyruvate	pyr_e
Threonine	thr_L_e
Tryptophan	trp_L_e
Myristic Acid	ttdca_e
Valine	val_L_e

alterations do not completely reverse the perturbation effect. Specifically, alteration of the media uptake fluxes does not restore the enforced virus optima back to the wild-type unconstrained optima. Summary of *in silico* media alteration results are shown in Appendix E.

Creation of mutated virus biomass objective functions

The effect of mutations of the viral genome(s) on the host-derived antiviral targets are further assessed through three mutation analyses: limited nucleotide variations (LNVs); virus-like particle mutations; and extensive nucleotide with structural component variations (ENVs).

For the LNVs, a total of 6000 mutant genomes were created each for CHIKV, DENV and ZIKV. These 6000 mutant genomes are comprised of 6 distinct groups, each which has a different number of point mutations randomly interspaced in the

nucleotide sequence of the genome: 1, 2, 3, 4, 5 or 10 nucleotide mutations at a single time. The resulting genomes were translated into the appropriate polyprotein for each virus, and new mutant VBOFs were created. These were individually analysed under the same host-derived enforcement methodology to assess the impact that point mutations may have on predicted antiviral targets. To assess whether the results for the point mutated viruses differ qualitatively or quantitatively, the average and standard deviation of the enforced optima (of the point mutated VBOFs) were analysed (Appendix E). In each case, for CHIKV, DENV and ZIKV, the set of reactions predicted to be effective against the point mutated VBOFs is qualitatively identical to that of the wild-type VBOFs. Quantitatively, the reaction-associated reduction of the wild-type VBOF lies within a single standard deviation of the average effect across all point mutations, for that viruses. In other words, the results are centred tightly around the original enforcement result for the wild-type VBOF.

To further assess the impact that virus mutation may have on the predicted antiviral targets, and to assess whether the results for the wild-type VBOF are statistically significant, a further set of mutants were created: Virus-Like Particles (VLPs) and ENV VBOFs. A visualisation of the variation in stoichiometric coefficients, for the LNV, VLP and ENV conditions, are shown for each virus in Appendix E.

A series of one-way analysis of variances (ANOVAs) on these mutants, compared to the wild-type and point mutation VBOFs, were conducted for each reaction in the model and the associated predictions.

For each virus, across most effective reactions (that is a reduction from the wild-type optima), it was identified that the biomass reduction associated with the predicted reactions is statistically significant when compared to their effects for the virus-like and ENV biomass functions. In essence, most of the antiviral targets predicted using this methodology are unique, both in terms of the quantitative effect on the virus (i.e. the reduction of virus optima) when compared across different virus species, but also when compared against randomly generated populations (which can be considered as virus-like and non-virus like entities). A summary of these statistics is available in Table. 3.6. This was calculated from looking at the variance between the viruses of the mutant category, against the original, wild-type viruses. There are a couple of exceptions only in the case of ZIKV, where the reactions mediated by cystathionine g-lyase (CYSTGL) and cystathio-

nine beta-synthase (CYSTS) did not show any significance in their effect under flux enforcement.

Table 3.6: Summary of Tukey test for Chikungunya (CHIKV), Dengue (DENV) and Zika (ZIKV) viruses. Comparisons were made between the aggregated group (WT+PM) of wild-type (WT) and limited nucleotide variation (LNV) viruses against the extensive nucleotide with structural component variations (ENV) and the WT and LNV of the other virus species. Frequency of single-reaction host-derived enforcements are stated, for the statistically significant ($p < 0.05$) and insignificant ($p \geq 0.05$) results.

(1) Virus	**(2) Condition or Virus Grouping**	**Frequency Tukey HSD Reject ($p < 0.05$)**	**Frequency Tukey HSD Accept ($p \geq 0.05$)**
CHIKV WT+LNV	CHIKV ENV	75	16
	DENV WT+LNV	75	75
	ZIKV WT+LNV	75	16
DENV WT+LNV	DENV ENV	65	0
	CHIKV WT+LNV	63	2
	ZIKV WT+LNV	59	6
ZIKV WT+LNV	ZIKV ENV	154	11
	CHIKV WT+LNV	145	20
	DENV WT+LNV	141	24

Further analysis with a post hoc Tukey test showed that most of the reactions identified as broad-targets in the host-derived enforcement analysis also showed significant differences in the magnitude of their effects among the three different viruses. In other words, while the reactions we highlight are not necessarily unique when comparing amongst CHIKV, DENV and ZIKV, their quantitative effects on virus production is significantly different for each species. Combined with the fact that our randomisation process maintained the key features of stoichiometric differences among the host and virus-like biomass functions, we highlight that the flux-perturbing effects of the identified reactions emerge from the core metabolic stoichiometric differences between host and the viruses.

3.4 DISCUSSION

In this chapter, a computational approach is presented that combines application of FBA (and variants) and FVA, with the development of integrated host-virus metabolic models, for antiviral prediction. This approach recovers known metabolic (reaction targeting) antiviral targets within an integrated human macrophage metabolic model, and predicts new potential targets against three viruses: CHIKV, DENV and ZIKV.

The predictions here are formed primarily of reactions that are involving nucleotides and amino acids. Further developing upon the previous chapter study §2.3.1, this host-derived enforcement analysis highlights the importance of the differences in host and virus metabolite usage and thus composition. As shown, many of the reactions predicted to be effective against any of the three viruses involve the metabolism of nucleotides. This can be explained by the relative ratios of the host- (macrophage) and virus-associated biomass objective functions (Table 3.7).

Table 3.7: Comparison of the amino acid (AA) and nucleotide (NTPS) biomass objective compositions and subsequent ratio of nucleotides to amino acids. Human macrophage, iAB-AMO-1410; Chikungunya virus, CHIKV; Dengue virus, DENV; Zika virus, ZIKV.

Biomass Function	**Proportion of biomass**		
	NTPS %	AA%	NTPS:AA
iAB-AMO-1410	3.78	86.45	0.04
CHIKV	7.26	92.74	0.08
DENV	13.11	86.89	0.15
ZIKV	12.87	87.03	0.15

Comparing the biomass objective functions in this way highlights the considerably higher proportion of nucleotide-associated biomass in the viruses than the host macrophage. This increased requirement for nucleotides propagates into the antiviral predictions; the host has a lower nucleotide level to satisfy than the virus, making nucleotide-producing and consuming reactions viable antiviral targets. This is also mirrored in the overlap with the metabolic-targeting antivirals identified from literature search (Leyssen et al. 2008, Khan et al. 2011, Hoffmann et al. 2011), whereby they almost exclusively target nucleotide associated reactions. These sequential, non-branching pathways simultaneously highlight why

nucleotide biosynthesis is a useful antiviral but non-viable host target. Both the host and the virus require these for their own production, and whilst the host utilises a smaller proportion of nucleotides (relative to its overall biomass), the topology of the nucleotide biosynthesis network does not allow for significant redundancies.

Unlike amino acid metabolites, where the reactions and pathways formed are intrinsically branched and their topology demonstrates multiple routes between precursors and the end amino acids, nucleotides do not exhibit the same type of network topology. Instead, the nucleotide metabolites are biosynthesised via a series of single-reaction chains, resulting in a chain of reactions that has reduce robustness, in terms of alternative network routes if a reaction were to be inhibited / altered, when compared to amino acids (Ravasz et al. 2002, Guimerà & Nunes Amaral 2005). This is save for a small set of salvage pathways which still require either externally supplied nucleotide precursors or catabolism of existing nucleotides. Subsequently, this issue surrounding the effects of network topology and the overlap of biomass precursors (between host and virus) highlights the advantages of the host-derived enforcement therapeutic approach over the reaction knockout approach, in terms of host viability and successful antiviral therapeutic delivery (Littler & Oberg 2005).

Both the single-reaction knockout and host-derived enforcement analysis highlight nucleotide associated reactions. However, the knockouts result in the reduction of host optima as well as the virus optima (in both cases the optima is reduced to zero). Not only does this demonstrate the level of entanglement between host and virus (and indeed the dependency of the virus on the host) but also the infeasibility of single-reaction knockouts as a viable antiviral therapeutic approach; this antiviral treatment strategy would result in large-scale host and virus damage which may perturb host viability and successful antiviral treatment. Host-derived enforcements predict antiviral reaction targets that can be perturbed to reduce virus optima whilst having minimal or no effect on the ability of the host to optimise. This approach appears advantageous, as it is able to directly combat the virus' production ability whilst allowing the host to remain viable until conclusion of the infection or identification and eradication/support by the host immune system.

The results of this analysis are in line with an integrated perspective that views the virus as an additional metabolic burden on the host cells, that could be met

or avoided by tinkering of host metabolic fluxes. The observed overlap between predicted reactions and known antiviral drugs gives confidence to this integrated modelling approach and highlights its potential as a prediction tool to guide experimental design. In addition, the model and virus sensitivity analysis gives confidence that this approach is highlighting antiviral targets that are not only virus-species specific, but also specific to virus organisms themselves (as opposed to all organisms containing the same viral biomass precursors). This can be especially useful in the case of new and emerging viruses for which limited clinical and experimental data may be available to inform drug target identification using existing compound databases (Wishart et al. 2008, Schomburg 2004).

The integrated stoichiometric metabolic modelling approach focuses on metabolic changes as a driver of virus production and does not consider factors associated with virus-host cell recognition, viral entry, lipid envelope production, and release (Timm & Yin 2012). Furthermore, the application of the linear optimisation on stoichiometric models (i.e. FBA and FVA) strictly assumes that host metabolism is at steady state, and thus prohibits analysis of the dynamics of cellular physiology. Such dynamics could be taken into account to a certain extent by imposing different flux constraints, which could be derived from proximal experimental data (Birch et al. 2012), through the development of simplified metabolic temporal models (Molenaar et al. 2009, Weiße et al. 2015), or by combining dynamics with linear optimisation on stoichiometric models (Birch et al. 2014, Mahadevan et al. 2002). Additionally, the extent of the missing information, such as genes, enzymes or reactions, in genome-scale stoichiometric models creates limitations on how much of the metabolic processes can be covered (Aurich & Thiele 2016).

Future efforts to improving model curation and standardisation (Zomorrodi & Segrè 2016) would open up the possibility of extensive analysis of host-virus pairings from a metabolic stance. Such modelling efforts would immensely benefit from a collection of appropriate, relevant experiment datasets. In particular, experimental analysis of cellular metabolic fluxes, as well as the determination of cellular uptake rates and metabolite requirements, can allow direct evaluation of the model. The presented findings already suggest that integrated host-virus models can highlight metabolic changes in the host and predict principal host metabolic processes that are linked to host-virus compositional mismatches and that can be used to combat virus production without altering host functions (Grantham 1974). In particular, analysis of extended flux enforcement strategies such as flux limitations on double reaction combinations might identify virus-specific drug

combinations (Littler & Oberg 2005). Combined with the future development of additional host-virus integrated models covering many cell and virus types can thus allow a fruitful route to the computational guiding of experimental antiviral drug discovery.

CHAPTER 4

Double-Reaction Perturbations of the Chikungunya, Dengue and Zika Virus Integrated Metabolic Models

4.1 INTRODUCTION

The previous chapter explored the effectiveness of single-reaction targeting as means of predicting potential antiviral therapeutics. In particular, a novel methodology was developed, "host-derived enforcements", that apply an additional constraint to the metabolic system that maintains host-optimality whilst perturbing virus production. We highlighted many targets that were mostly confined to the *de novo* nucleotide biosynthetic pathways. CHIKV, DENV and ZIKV were all affected by these reactions, albeit quantitatively different, and the average result was a small reduction in virus optima: approximately 70-80% of the wild-type optima.

Given the interplay between the virus and host-metabolic network (Sanchez et al. 2015) and the relative lack of metabolic-based antivirals for RNA viruses outside of nucleotide biosynthesis (Leyssen et al. 2008), a multi-reaction approach (in the context of host metabolism) could provide fruitful antiviral theraputics. Realising this systems-level understanding of host-viral metabolic interactions and how to effectively perturb virus production requires the development of a computational approach that not only effectively predicts multi-reaction antiviral therapies, but explores the impact on host-virus metabolic resource allocation. Previous experimental and clinical studies have highlighted the effectiveness of both multiple target and multiple drug antiviral approaches in respiratory virus (Hayden 1996)

and Hepatitis C viral infections (Poordad et al. 2013).

Metabolic systems are highly complex, branched networks that contain multiple pathways, compartments and subsystems (Schuster et al. 2000, Hatzimanikatis et al. 2005). Although simplified in their representation, metabolic networks display a level of redundancy in the production of biomass components (Wang & Zhang 2009). This redundancy, from an evolutionary and survivability aspect, ensures that an organism can still produce the required biomass components even when the metabolic network is subject to damage or distress (Almaas et al. 2005). It is this property of metabolic networks that potentially makes multiple-reaction targeting a more optimal solution for antiviral development: since biomass metabolites can be created using more than one pathway, the corresponding antiviral treatment would too require more than one control point (in this case reactions).

To better explore the possibility of multiple metabolic perturbations as an antiviral therapeutic, here we develop an extension to our preexisting methodology and constrain two control points [reactions] of the metabolic network against CHIKV, DENV and ZIKV.

4.2 METHODS

4.2.1 Integrated Host-Virus Metabolic Model Selection and Creation

Three virus species were selected, as previously defined in §2.2.1, Table 2.1: Chikungunya virus (CHIKV), dengue virus (DENV) and zika virus (ZIKV). VBOFs and integrated host-virus metabolic models were created for each individual virus as described previously in §2.2.2.

4.2.2 Double Reaction Perturbations of Integrated Host-Virus Metabolic Models

As explored in previous chapters 2 and 3, reaction perturbations represent a potential avenue for the discovery of antiviral targets. however, these have only been considered in terms of a single reaction. In an expansion to this methodology, combinations of two model reactions, called 'double perturbations', are explored under two conditions: double reaction knockouts and double reaction host-derived

enforcements. Toy models / schematics for both conditions are shown in Fig. 4.1(A) and (B) respectively.

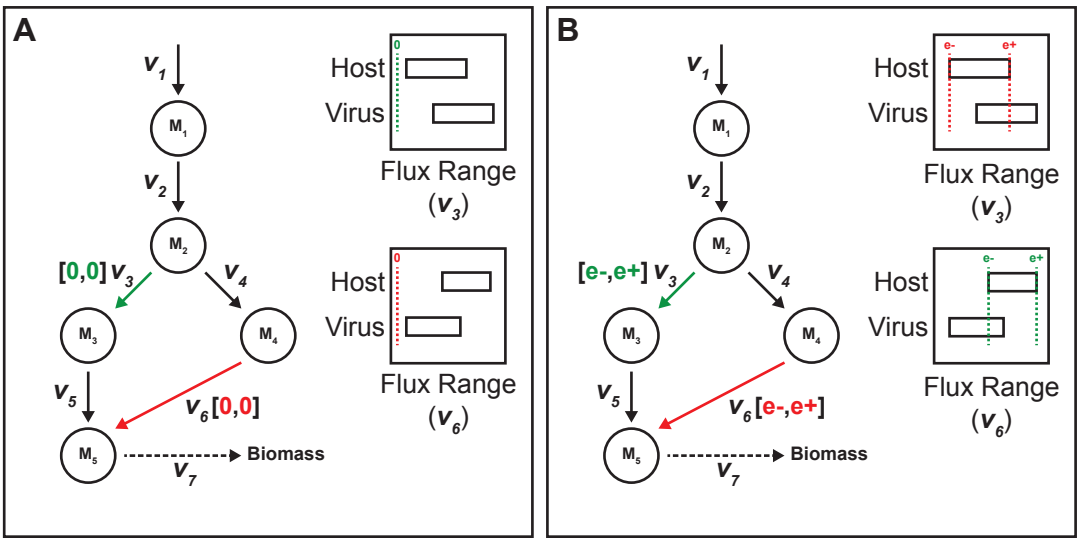


Figure 4.1: Toy model of flux balance analysis optimisation problems for (**A**) double-reaction knockouts and (**B**) double-reaction host-derived enforcements. Toy networks show metabolites (M) as nodes and reactions as edges, with associated reaction flux (V) marked. Flux ranges are shown for the enforced reactions, where the minimum ($e-$) and maximum ($e+$) enforcement bounds are highlighted.

Double Reaction Knockout Analysis

Knockout analysis considers the effect of systematically setting individual reaction fluxes to zero, and then attempting to maximise the integrated host-virus metabolic model for virus production, previously described as a single-reaction linear optimisation problem in §3.2.2. Here, double-reaction knockouts are considered, to see if there are combinations of reactions that, when their flux bounds are set to zero, can differentially affect virus optima. To implement this, an additional inequality is added to the single-reaction knockout linear optimisation problem (Eq. 3.1), yielding the double-reaction knockout linear optimisation problem (Eq. 4.1).

$$\begin{aligned} \text{maximise} \quad & Z_{ko} = \mathbf{c}^T \cdot v \\ \text{subject to} \quad & S \cdot v = 0 \\ & lb < v < ub \\ & 0 < v_i < 0 \\ & 0 < v_j < 0 \end{aligned} \tag{4.1}$$

where indexation i and j are over all reactions v in the model systematically and the associated lower (lb) and upper (ub) bounds are simultaneously set to zero. Note that under this condition, reaction $i \neq j$, as this is equal to a single-reaction knockout. This model is then optimised for VBOF, yielding the virus knockout optima (Z_{ko}). A reaction knockout is considered 'effective', in terms of its potential as an antiviral target, if $Z_{ko} < Z_{wt}$.

Double Reaction Host-Derived Enforcement Analysis

As previously described in §3.2.2, an integrated host-virus metabolic model can be additionally constrained in such a way that host-viable flux ranges are enforced whilst the model is optimised for virus production. Similar to the reaction knockout condition, and additional inequality is added to the single-reaction host-derived enforcement linear optimisation problem (Eq. 3.5), yielding the double-reaction host-derived enforcement linear optimisation problem (Eq. 4.2).

$$\begin{aligned} \text{maximise} \quad & Z_e = \mathbf{c}^T \cdot v \\ \text{subject to} \quad & S \cdot v = 0 \\ & lb < v < ub \\ & e_i^- < v_i < e_i^+ \\ & e_j^- < v_i < e_j^+ \end{aligned} \tag{4.2}$$

where indexation i and j are over all reactions v in the model systematically and the associated lower (lb) and upper (ub) bounds are set to the calculated lower (e^i) and upper (e^j) enforcement values. Note that under this condition, reaction $i \neq j$, as this is equal to a single-reaction enforcement. This model is then optimised for VBOF, yielding the virus knockout optima (Z_e). A reaction knockout is considered 'effective', in terms of its potential as an antiviral target, if $Z_{ko} < Z_{wt}$.

4.2.3 Epistatic interactions

The epistatic interactions between the single-reaction (previous study) and double-reaction host-derived enforcement targets were compared to assess whether reactions, when paired, act in a synergistic manner (that is, the effect they have on the virus host-enforced optima Z_e is more than would be expected from their effects in the single-reaction perturbations). For this method, we quantify a negative epistatic effect as a desired epistatic effect: the combination of these reactions reduces the virus optima more than would be predicted and therefore is highlighted as a beneficial (antiviral) target.

The epistatic interactions (ε) between pairs of reactions are calculated by:

$$\varepsilon = f_{AB} - f_A f_B \tag{4.3}$$

where AB are the double-reaction enforcement virus optima, and A and B are the single-reaction enforcement virus optima for the reactions A,B in the double-enforcement condition. If both reactions are identical ($A = B$) then ε cannot be calculated and (for the purposes of the results) recorded as 'NaN'.

4.3 RESULTS

4.3.1 Double-Reaction Enforcements Highlight Multiple Metabolic Network Targeting Points

In order to assess the impact that combining multiple reaction pairs, under the conditions of knockouts and host-derived enforcements, may have on virus production, a double-reaction methodology was created and implemented for the iHVMs of CHIKV, DENV and ZIKV associated metabolic models.

The integrated HVMs identified multiple pairs of reactions that, compared to each reaction in the single-enforcement condition from the previous study, result in a synergistic effect that results in a more substantial than expected decrease in virus optima. In this way, we define a "beneficial" synergy (concerning antiviral effect) as a negative value. Many of these reaction pairs when grouped by subsystems are effective in the context of amino acid metabolism. For CHIKV and DENV, combinations of nucleotide associated reactions do not lead to the synergistic ef-

fects that are seen with the combination of amino acid associated reactions. An exception of this is the case of ZIKV, where a mixture of nucleotide and amino acid reactions cause increased perturbation to the ZIKV optima, beyond what would be expected from a combination of the single reaction perturbations.

4.3.2 Double Host-Enforcements: Chikungunya Virus

From the subset of reactions that are effective against CHIKV in the double-reaction condition, we highlight reactions that exhibit a synergistic effect when paired together. As shown in Fig. 4.2, there a few sets of reaction pairs that exhibit a large negative value of synergy ($\varepsilon \leq -0.6$). This indicates that when these reactions are paired together, the effect on the virus (in terms of optima reduction) is greater than what would be predicted from the single-reaction perturbations alone. The absolute values of these double-reaction host-derived enforcements, in terms of reduction of virus optima (compared to wild-type), are shown for CHIKV in Fig. 4.3.

Reactions that have a positive synergy, indicating that pairing of the reactions does not reduce the virus optima to a level that would be predicted from the single-reaction results, are also highlighted. The majority of these reactions are associated with the interactions of both the Nucleotides and Pentose Phosphate Pathway subsystems, with other metabolic reaction subsystems.

Reactions are attributed to a model subsystem, and these can also be further categorised based upon the major metabolic junction (defined in the context of a single metabolite) that these subsystems are associated with. In the case of these CHIKV-associated negative synergy reactions, they broadly fall into two categories: cysteine related and threonine related.

Concerning cysteine, we observe that restricting the anabolic reactions and increasing the catabolic reactions reduces its availability for virus production. Cysteine is considered a non-essential amino acid and is synthesised (along with glycine and serine) from glycolytic intermediates. In that sense, we can target both sides of metabolism (anabolic and catabolic) to perturb the virus production.

These identified reactions predict that downregulating the supply of precursors for cysteine biosynthesis such as Cystathionine (CYST-L), while simultaneously increasing the flux of cysteine-consuming reactions, result in a decrease of CHIKV optima. The combination of these reactions pairs causes cysteine to become a

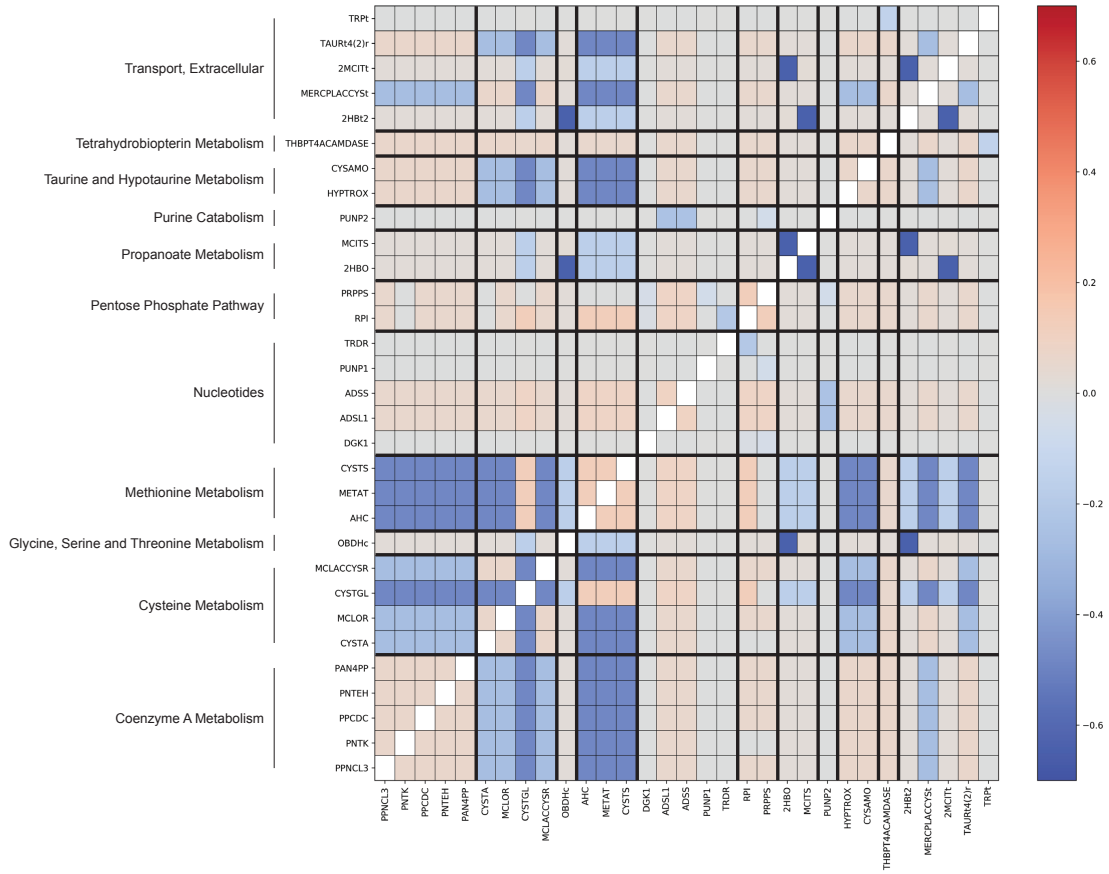


Figure 4.2: Exhaustive comparison of Chikungunya virus double-reaction host-derived enforcement epistatic interactions, grouped by subsystem. Colour bar indicates the epistatic interaction value; greater antiviral effect (blue) to lesser antiviral effect (red) than would have been predicted from the single-reaction enforcements alone. Reaction IDs are shown on the x- and y-axis, combining to form a double-reaction pair. Pairings of identical reactions are highlighted by an unfilled square. Subsystems are labeled on the y-axis. Reaction IDs with full names are given in the original model publication (Bordbar et al. 2010).

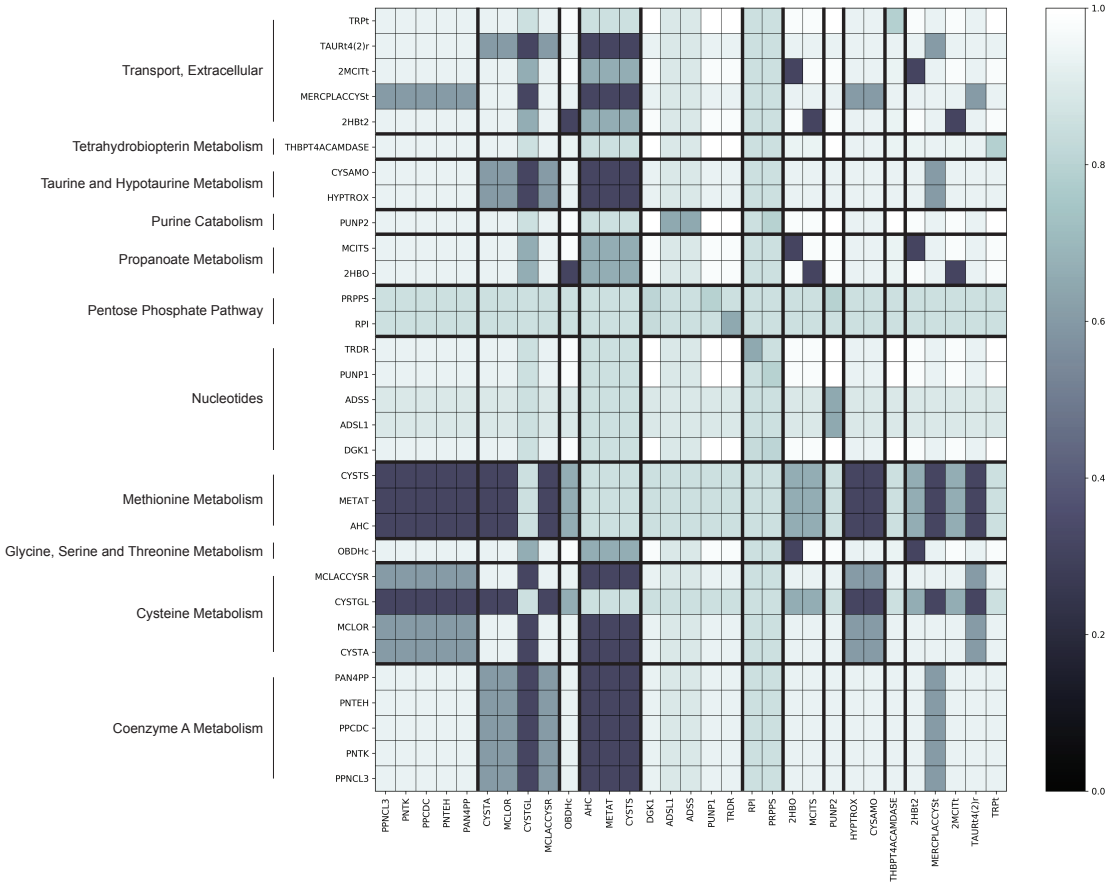


Figure 4.3: Exhaustive comparison of Chikungunya virus double-reaction host-derived enforcement reactions, grouped by subsystem. Colour bar indicates the proportion of original wild-type virus optima when two reactions simultaneously undergo host-derived enforcement; greater optima reducing effect (dark blue; virus optima is 0% of wild-type) to lesser antiviral effect (white; virus optima is 100% of wild-type). Reaction IDs are shown on the x- and y-axis, combining to form a double-reaction pair. Where pairings of reactions are identical, the value shown corresponds to the effect of that reaction under the single-reaction host-derived enforcement analysis. Subsystems are labeled on the y-axis. Reaction IDs with full names are given in the original model publication (Bordbar et al. 2010).

limited resource which the virus requires for optimal production (Fig. 4.4). As has been shown with the comparison between host and viral usage of amino acids, CHIKV has an almost 2-fold increased requirement for cysteine when compared to the host (Appendix F). This mismatch of usage potentially allows for a reduction in the availability of cysteine, something that the host can tolerate (and remain optimal) but CHIKV cannot. Ultimately, the combination of pairs upregulated (+) and downregulated (-) reactions appears to show the greatest antiviral effect, as is discussed throughout these results.

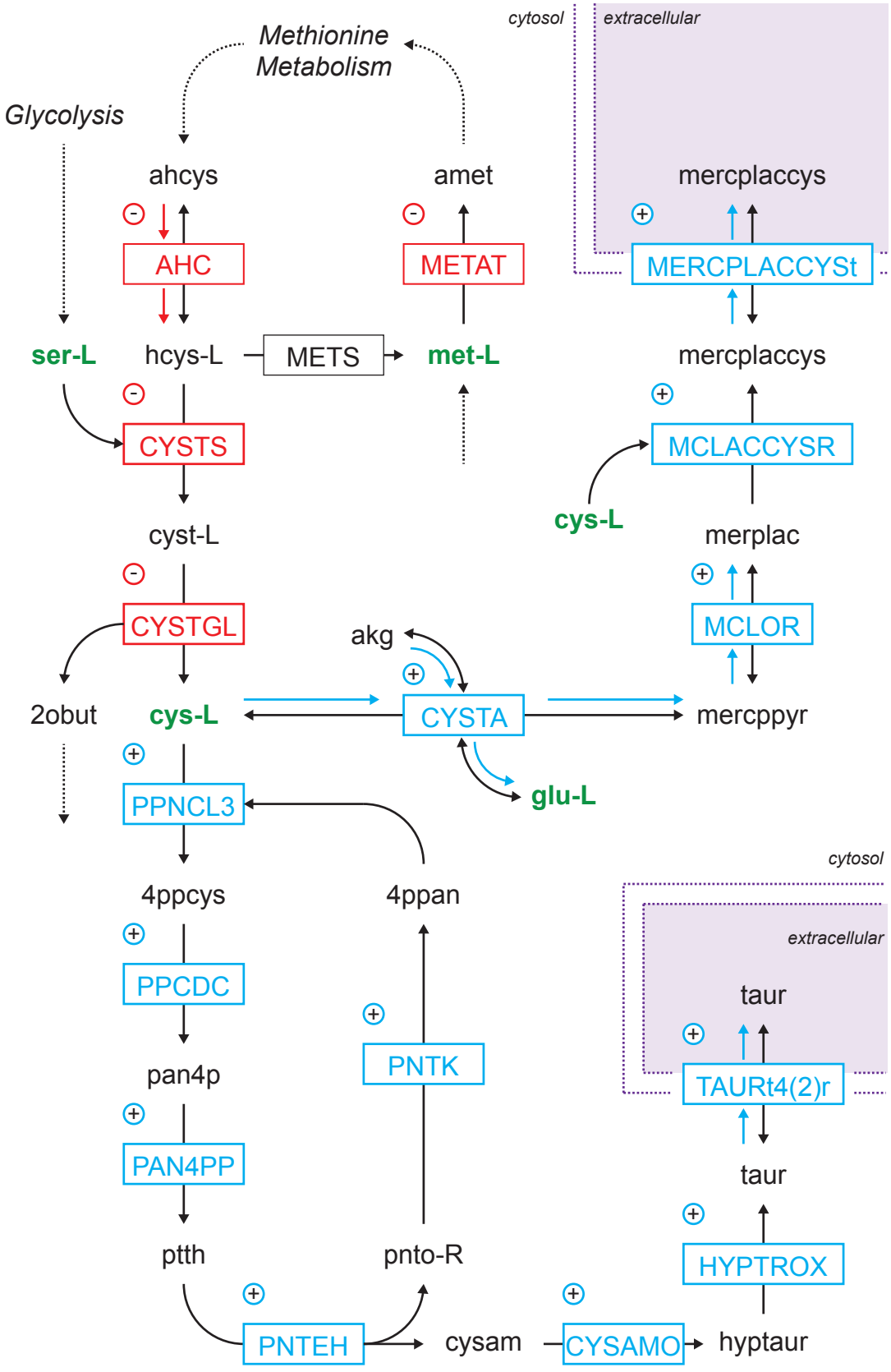

cytosol
extracellular
Methionine Metabolism
Glycolysis
ahcys
amet
mercplaccys
AHC
METAT
MERCPLACCYSt
ser-L
hcys-L
METS
met-L
mercplaccys
CYSTS
MCLACCYSR
cys-L
cyst-L
merplac
CYSTGL
MCLOR
akg
2obut
cys-L
CYSTA
mercppyr
glu-L
PPNCL3
cytosol
extracellular
4ppcys
4ppan
taur
PPCDC
PNTK
TAURt4(2)r
pan4p
taur
PAN4PP
HYPTROX
ptth
pnto-R
PNTEH
cysam
CYSAMO
hyptaur

Figure 4.4 ***(previous page)*****:** Chikungunya virus double-reaction host-derived enforcements that are associated with cysteine (CYS-L) metabolism. Synergistic reaction pairs are formed between any (■) and (■) reactions, when simultaneously enforced. Reactions can be upregulated (+) or downregulated (-). Black arrows (➔) denote reaction directionality, and coloured arrows indicated the directionality of primary (➔) and secondary (➔) reaction pair members if reaction is bidirectional. Virus-associated biomass precursors are highlighted in green. Model compartment boundaries are indicated by dashed line (- - -). Reaction and metabolite abbreviations are expanded in the original model publication (Bordbar et al. 2010).

There are also reactions highlighted that are associated with the catabolism of 2-Oxobutanoate (2OBUT). In this case, pairs of reactions that catabolise 2OBUT are upregulated and thus increase the demand on 2OBUT (Fig. 4.5). 2OBUT is a product in the biosynthesis of cysteine, but also from the catabolism of threonine. Threonine is identified as an amino acid that is used almost 3-fold more in CHIKV composition than the host.

Similar to the case with cysteine, the pairing of reactions that are associated with the catabolism of 2OBUT cause a perturbation on the virus optima. Specifically, by upregulating the catabolism of 2OBUT to 2-Hydroxybutyrate (2HB) and Propanoyl-Coa (PPCOA) simultaneously, a demand is created within the metabolic network for 2OBUT synthesis. This can be satisfied either through increased cysteine biosynthesis, which may be of potential benefit to CHIKV due to its high biomass requirement for cysteine or through increased catabolism of threonine. In the latter case, while this would increase the supply of 2OBUT, it would decrease the availability of threonine to the virus optimised network. As this has been identified as a significant component of the CHIKV biomass, this results in an overall decrease of virus production within this constrained network.

Regarding threonine, we observe that increasing the catabolic reactions reduces the availability of threonine for virus production. Unlike cysteine, threonine is an essential amino acid and cannot be synthesised *de novo*. These examples, in particular those that are associated with amino acids that are biosynthesised *de novo*, highlight how the combined constraining of the in- and ef-fluxes of amino acid associated metabolites can create highly effective, synergistic perturbations that can reduce the ability of CHIKV to optimise within the [constrained] metabolic network.

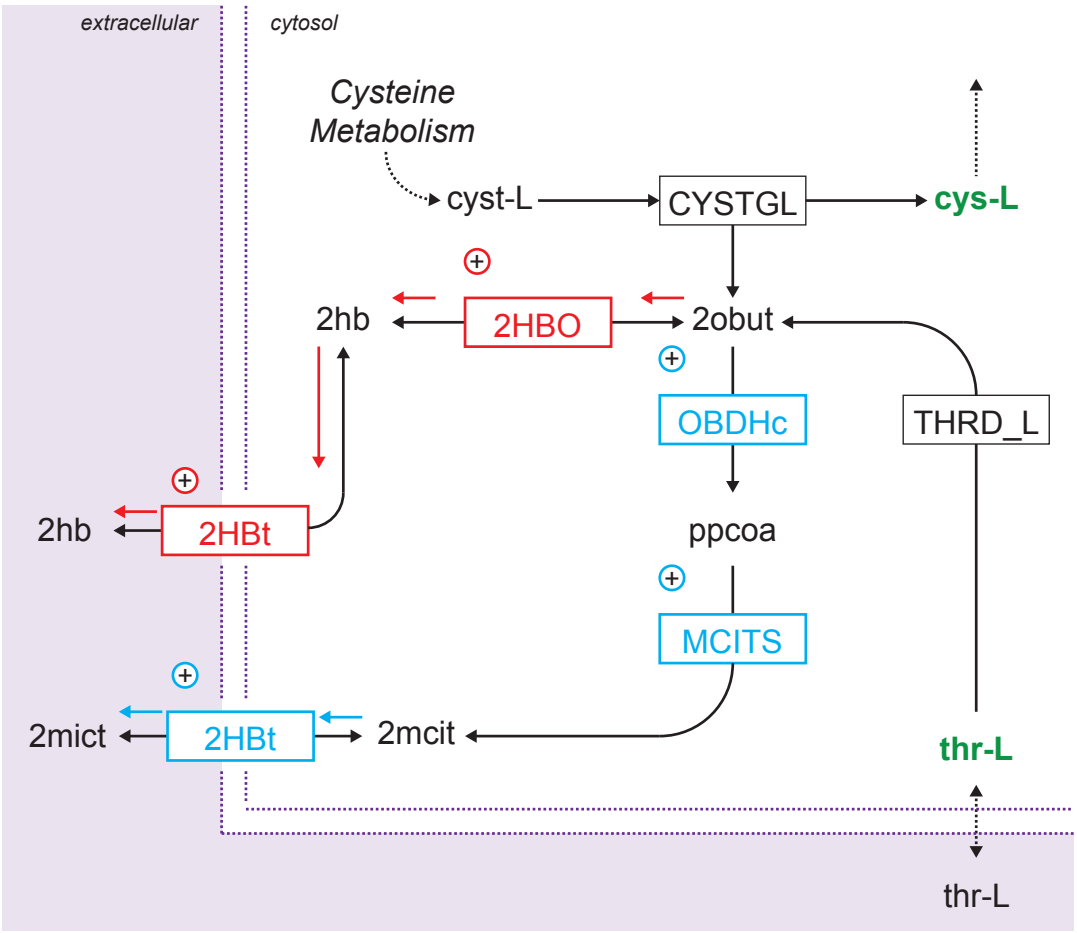


Figure 4.5: Chikungunya virus double-reaction enforcements associated with threonine metabolism. Synergistic reaction pairs are formed between any (■) and (■) reactions, when simultaneously enforced. Reactions can be upregulated (+) or downregulated (-). Black arrows (➔) denote reaction directionality, and coloured arrows indicated the directionality of primary (➔) and secondary (➔) reaction pair members if reaction is bidirectional. Virus-associated biomass precursors are highlighted in green. Model compartment boundaries are indicated by dashed line (- - -). Reaction and metabolite abbreviations are expanded in the original model publication (Bordbar et al. 2010).

4.3.3 Double Host-Enforcements: Dengue Virus.

From the subset of reactions that are effective against DENV in the double-reaction conditions, we highlight a similar set of synergistic reactions to that found in CHIKV (Fig. 4.6). The absolute values of these double-reaction host-derived enforcements, in terms of reduction of virus optima (compared to wild-type), are shown for DENV in Fig. 4.7.

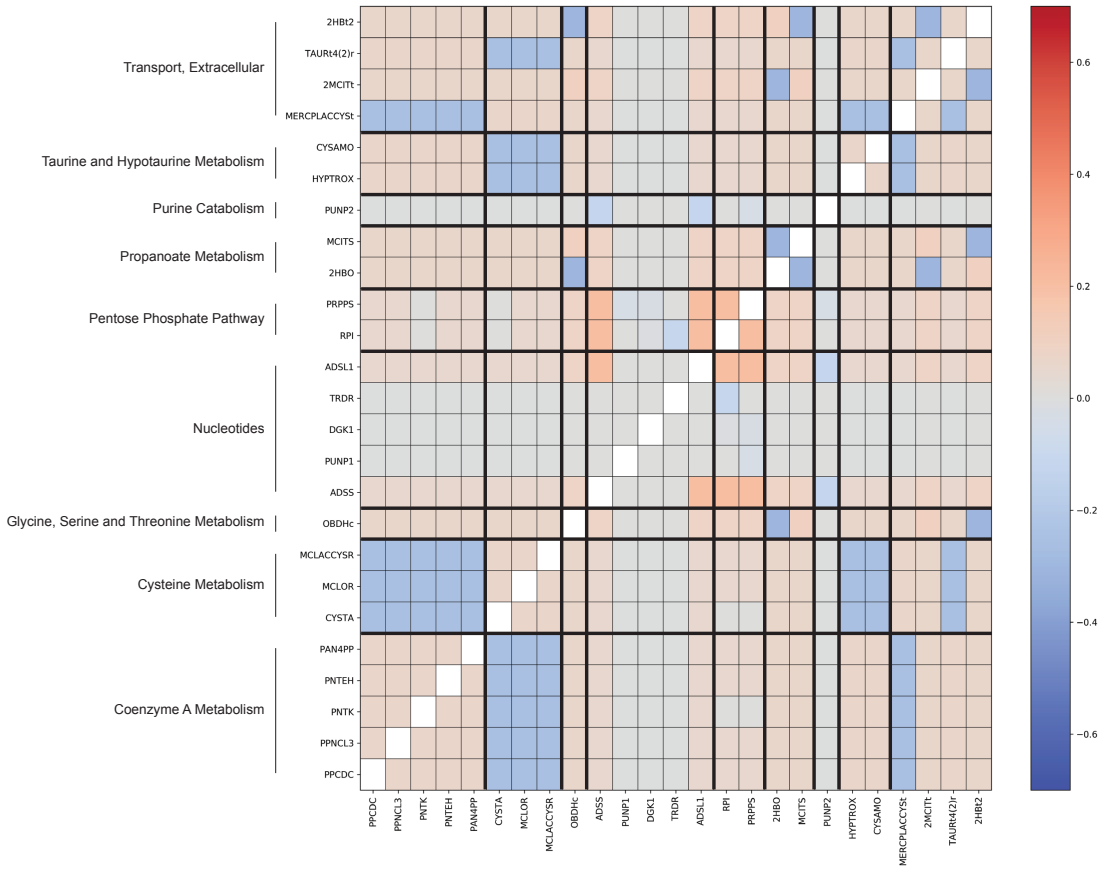


Figure 4.6: Exhaustive comparison of Dengue virus double-reaction host-derived enforcement epistatic interactions, grouped by subsystem. Colour bar indicates the epistatic interaction value; greater antiviral effect (blue) to lesser antiviral effect (red) than would have been predicted from the single-reaction enforcements alone. Reaction IDs are shown on the x- and y-axis, combining to form a double-reaction pair. Pairings of identical reactions are highlighted by an unfilled square. Subsystems are labeled on the y-axis. Reaction and metabolite abbreviations are expanded in the original model publication (Bordbar et al. 2010).

Notable exceptions are that for DENV we no longer find that cysteine anabolism is a viable target, that is, limiting the supply of cysteine biosynthesis precursors no longer causes a reduction in DENV optima. From the comparison between host

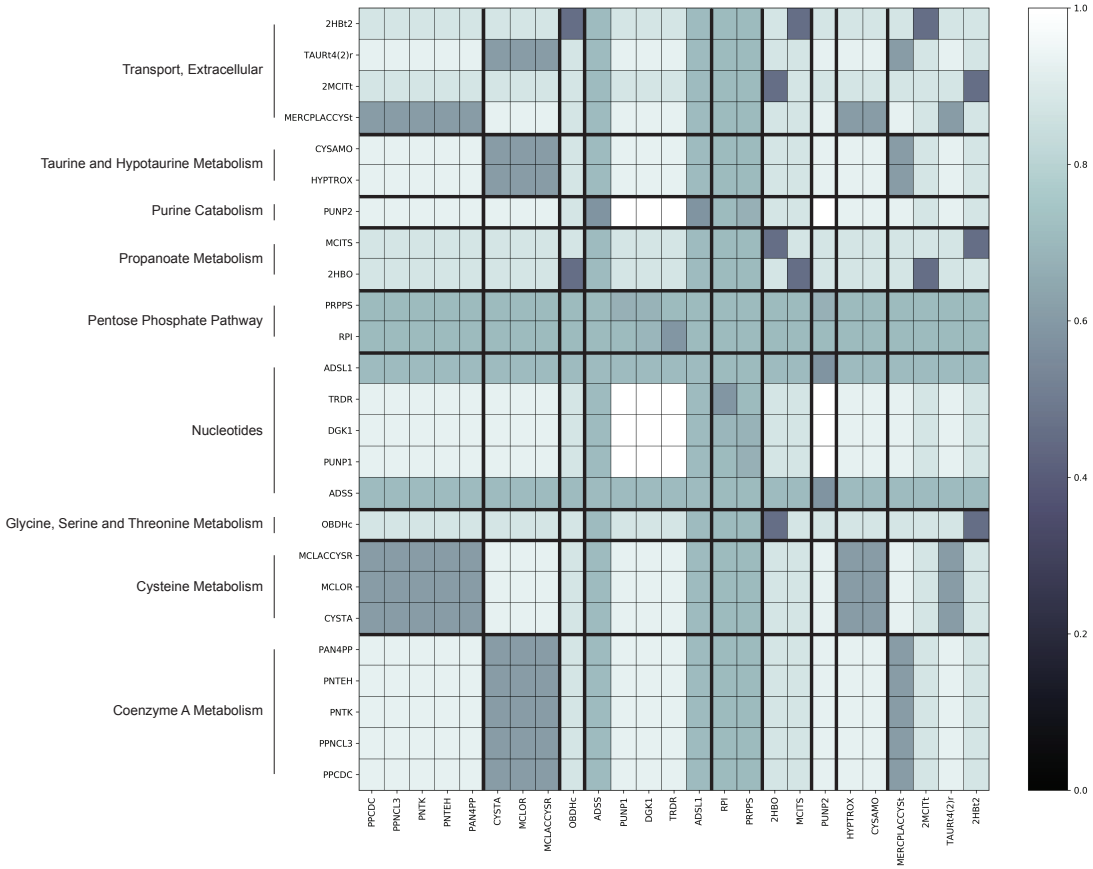


Figure 4.7: Exhaustive comparison of Dengue virus double-reaction host-derived enforcement reactions, grouped by subsystem. Colour bar indicates the proportion of original wild-type virus optima when two reactions simultaneously undergo host-derived enforcement; greater optima reducing effect (dark blue; virus optima is 0% of wild-type) to lesser antiviral effect (white; virus optima is 100% of wild-type). Reaction IDs are shown on the x- and y-axis, combining to form a double-reaction pair. Where pairings of reactions are identical, the value shown corresponds to the effect of that reaction under the single-reaction host-derived enforcement analysis. Subsystems are labeled on the y-axis. Reaction IDs with full names are given in the original model publication (Bordbar et al. 2010).

and virus compositions, it is clear that cysteine is proportionally equal in both the host and virus biomass objectives (Appendix B and C). In this sense, this means cysteine cannot be a target for cross-constraining. However, we observe that methionine is utilised roughly 2-fold more in the DENV biomass than the host's biomass function, and thus, displays several synergistic pairs that can be constrained.

Methionine, an essential amino acid, is involved in the biosynthesis of other cellular proteins as well as being a component of both viral and host biomass objective functions. The methionine derivative S-adenosyl methionine (AMET) is also involved in cysteine biosynthesis. Methionine supply is therefore dependent on the anabolic activity of cysteine. We identify reactions with the highest levels of synergy are associated with the cysteine and cysteine-linked coenzyme A metabolic pathways, that are associated with the metabolism of Methionine and its metabolic derivatives. Whereas in CHIKV, the most effective identified reaction pairs were a combination of an upregulated and downregulated reaction, in the case of DENV we highlight only pairs of upregulated reactions.

Specifically, pairings of CYSTA and PPNCL3 (both beginnings of sequential chains of reactions) when simultaneously upregulated results in decreased DENV optima (Fig. 4.8). The proposed mechanism for this is that through the upregulation, there is an increased demand for cysteine synthesis. This, in turn, requires that methionine (considered critical in the virus biomass composition due to its relatively high proportional usage) be catabolised into the derivatives required for cysteine synthesis. Ultimately this limits the amount of methionine available for virus biomass production, thus reducing the virus optima. In other words, there appears to be a trade-off between methionine and cysteine, which is "tipped against" the virus through the appropriate metabolic constraints.

The pairing of 2obut associated reactions is similar to CHIKV. Threonine is also identified as an amino acid that is present in a higher proportion of DENV biomass than in the host. Therefore, the proposed mechanism is the same as that of CHIKV, in that threonine becomes a limiting resource for virus biomass production.

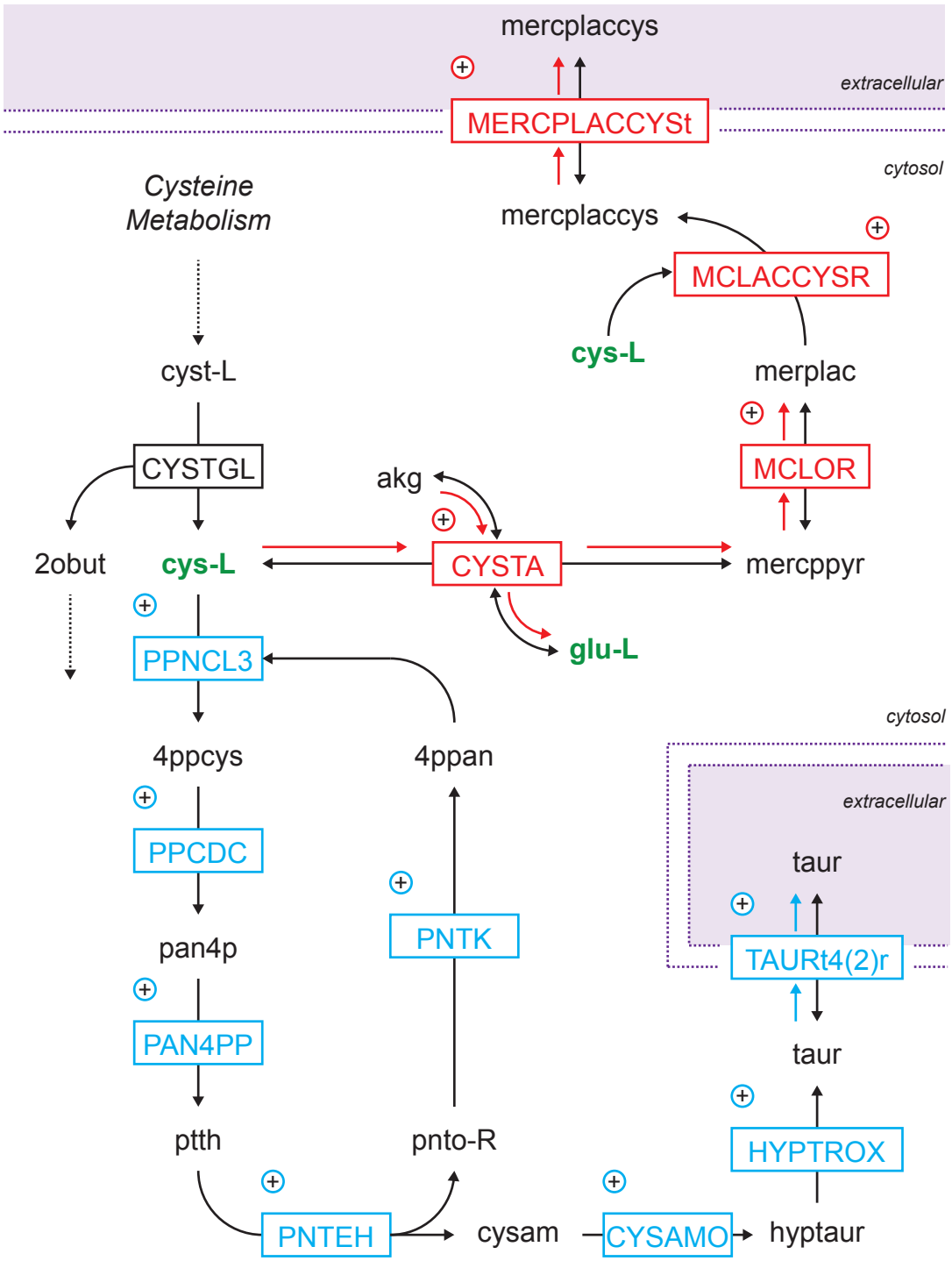


Figure 4.8: Dengue virus double-reaction enforcements associated with cysteine and methionine metabolism. Synergistic reaction pairs are formed between any (■) and (■) reactions, when simultaneously enforced. Reactions can be upregulated (+) or downregulated (-). Black arrows (➔) denote reaction directionality, and coloured arrows indicated the directionality of primary (➔) and secondary (➔) reaction pair members if reaction is bidirectional.
(continued on next page...)

Figure 4.8 ***(previous page)*****:** Virus-associated biomass precursors are highlighted in green. Model compartment boundaries are indicated by dashed line (- - -). Reaction and metabolite abbreviations are expanded in the original model publication (Bordbar et al. 2010).

4.3.4 Double Host-Enforcements: Zika Virus.

The double-enforcement conditions for ZIKV, which are effective in reducing virus optima and show synergistic effects, do not follow the same pattern as in CHIKV and DENV (Fig. 4.9). Absolute values of these double-reaction host-derived enforcements, in terms of reduction of virus optima (compared to wild-type), are shown for in Fig. 4.10.x As previously discussed, in these cases only reactions that are immediately associated with amino acids (or their metabolism) were directly identified. These identified amino acids are also highlighted as being differentially utilised between the viruses and the host biomass functions.

However, in the case of ZIKV, we find that the double-reaction enforcements that have the most substantial synergistic effect include both nucleotide and amino acid associated reactions. Broadly, we categorise the double-enforcement conditions into three groups: amino acid only; nucleotide and amino acid interplay; and nucleotide only.

As has been discussed previously for the CHIKV and DENV results, the differences in host and virus biomass composition play a substantial role in the predicted antiviral targets; this is particularly interesting for the ZIKV antiviral targets due to the prediction of nucleotide associated antiviral targets. ZIKV has a high requirement for glycine, an essential amino acid in the biosynthesis of nucleotides. While DENV also has a higher-than-host usage of glycine (CHIKV is similar to host) its requirement for serine and cysteine, two amino acids that are also derived from the same glycolytic intermediate as glycine, are lower / the same as the host (respectively). ZIKV, however, has a higher demand for serine than DENV, so the increased demand of multiple amino acids from the same precursor has potential downstream effects on both amino acid and nucleotide biosynthesis reactions.

Nucleotide associated reactions, therefore, represent an effective (and synergistic) target for ZIKV when compared to CHIKV and DENV. In particular, we find that a combination of purine and pyrimidine catabolic reactions cause a decrease in the virus optima. More specifically, by increasing the production of deoxyguano-

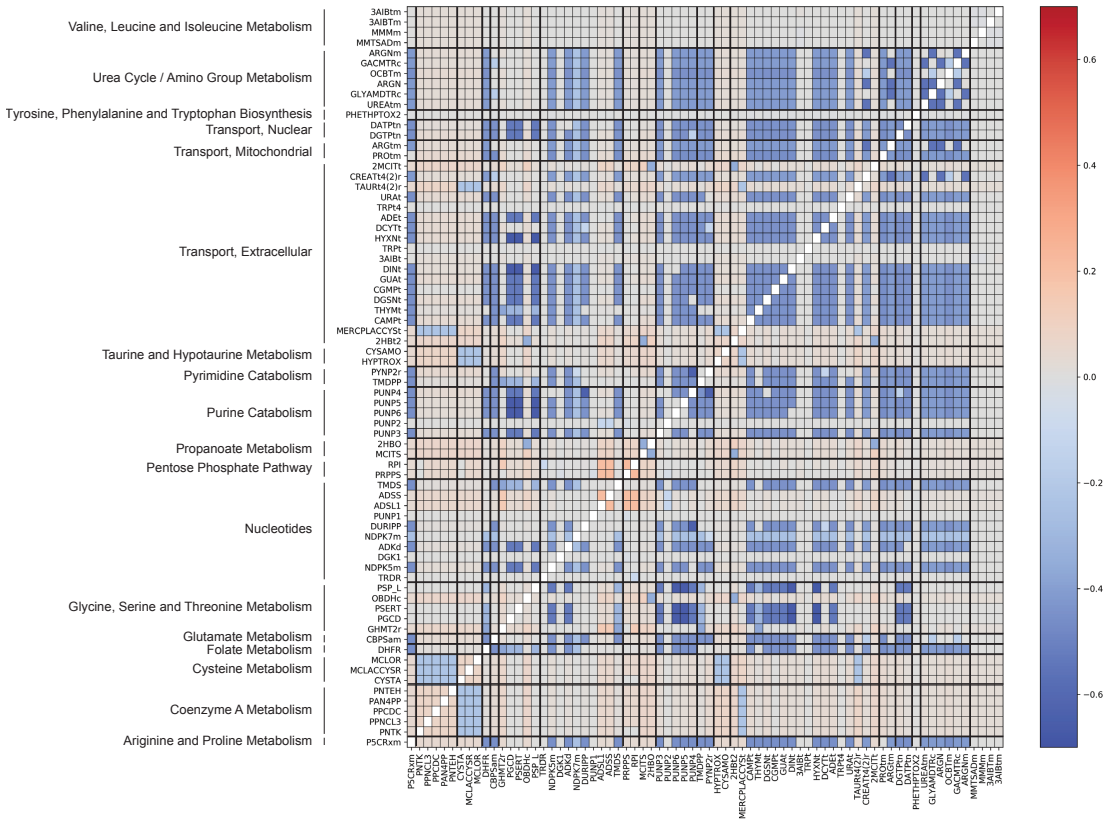


Figure 4.9: Exhaustive comparison of Zika virus double-reaction host-derived enforcement epistatic interactions, grouped by subsystem. Colour bar indicates the epistatic interaction value; greater antiviral effect (blue) to lesser antiviral effect (red) than would have been predicted from the single-reaction enforcements alone. Reaction IDs are shown on the x- and y-axis, combining to form a double-reaction pair. Pairings of identical reactions are highlighted by an unfilled square. Subsystems are labeled on the y-axis. Reaction and metabolite abbreviations are expanded in the original model publication (Bordbar et al. 2010).

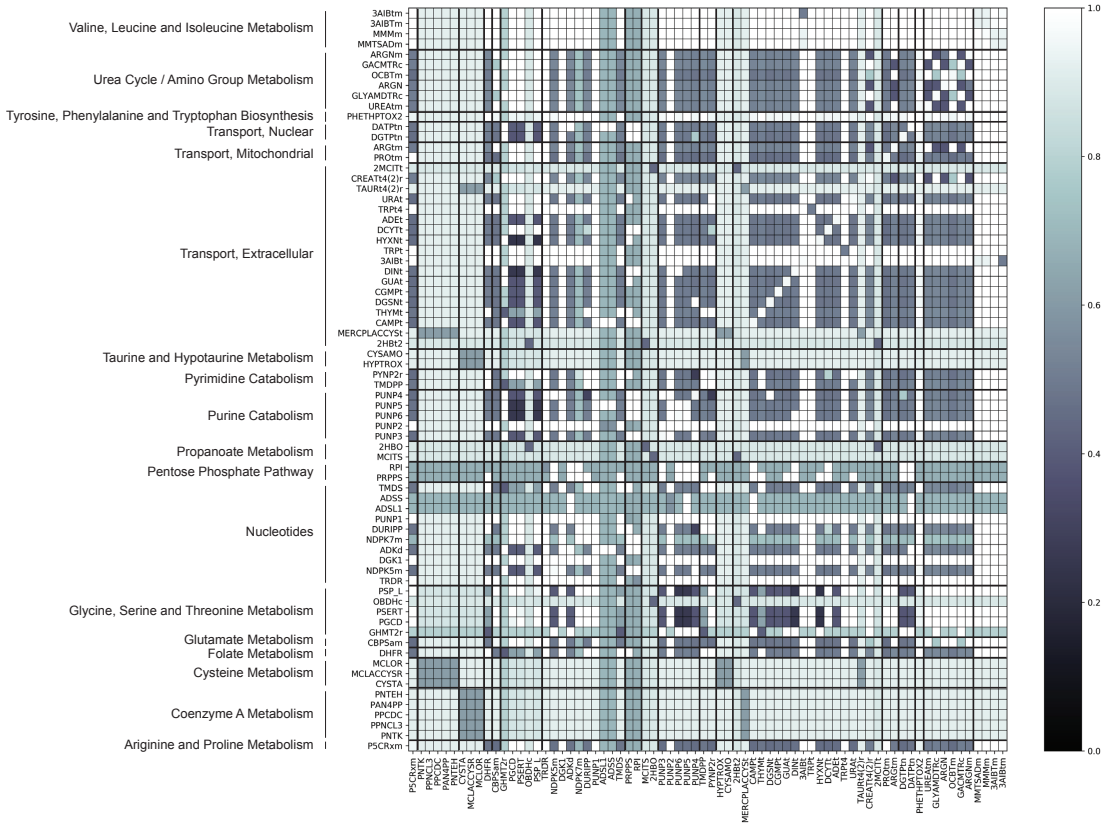


Figure 4.10: Exhaustive comparison of Zika virus double-reaction host-derived enforcement reactions, grouped by subsystem. Colour bar indicates the proportion of original wild-type virus optima when two reactions simultaneously undergo host-derived enforcement; greater optima reducing effect (dark blue; virus optima is 0% of wild-type) to lesser antiviral effect (white; virus optima is 100% of wild-type). Reaction IDs are shown on the x- and y-axis, combining to form a double-reaction pair. Where pairings of reactions are identical, the value shown corresponds to the effect of that reaction under the single-reaction host-derived enforcement analysis. Subsystems are labeled on the y-axis. Reaction IDs with full names are given in the original model publication (Bordbar et al. 2010).

sine and deoxyuridine from guanine and uridine (respectively), the supply of the associated RNA nucleotides which are required for virus biomass production is limited. This is achieved through the simultaneous upregulation of purine-nucleoside phosphorylases (deoxyguanosine-associated) with pyrimidine-nucleoside phosphorylases (uracil- and deoxyuracil-associated) causing a constraint on the precursors and intermediates required for RNA nucleotide synthesis and incorporation into the virus biomass (Fig. 4.11). In essence, as has been predicted for the other viruses, a combination of simultaneously regulating the in- and ef-flux of nucleotide biosynthetic intermediates creates constraints that provide antiviral action.

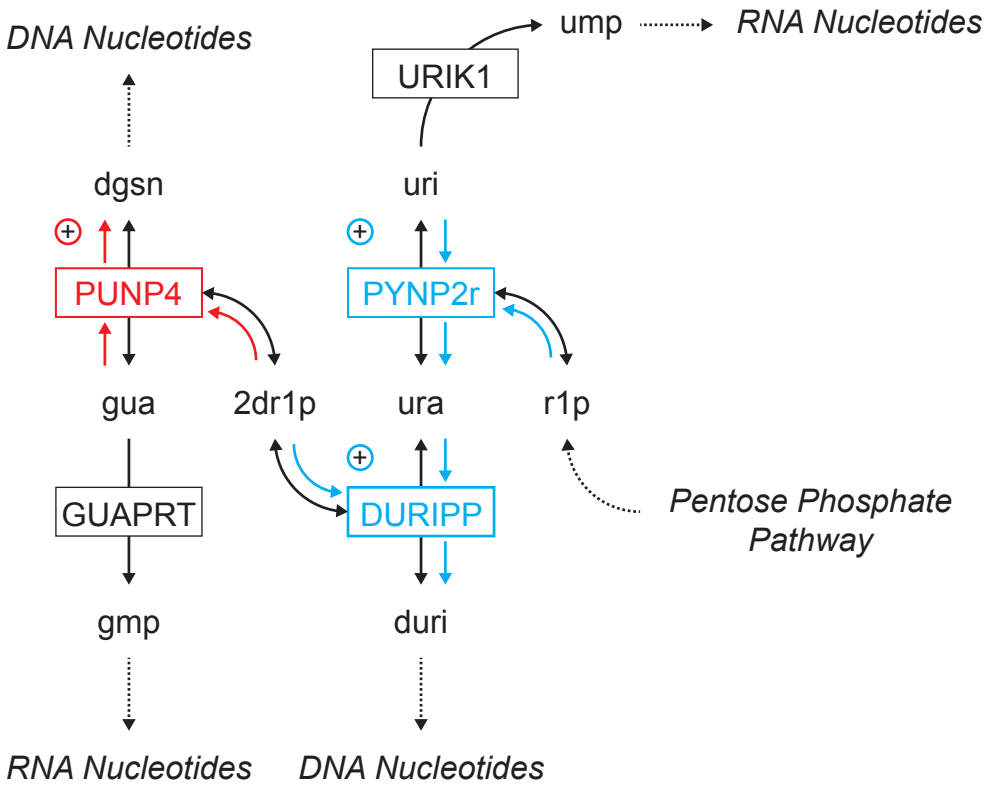


Figure 4.11: Zika virus double-reaction enforcements associated with pyrimidine and purine metabolism. Synergistic reaction pairs are formed between any (■) and (■) reactions, when simultaneously enforced. Reactions can be upregulated (+) or downregulated (-). Black arrows (➔) denote reaction directionality, and coloured arrows indicated the directionality of primary (➔) and secondary (➔) reaction pair members if reaction is bidirectional. Virus-associated biomass precursors are highlighted in green. Model compartment boundaries are indicated by dashed line (- - -). Reaction and metabolite abbreviations are expanded in the original model publication (Bordbar et al. 2010).

Some nucleotide-amino acid associated reaction pairs were also identified as ef-

fective against ZIKV. Specifically, the combination of serine and serine-precursor synthesis reactions with purine catabolism reactions results in decreased virus optima (Fig. 4.12). While ZIKV proportionally uses less serine than that of the host, its requirement for glycine is much higher. The serine synthesis reactions provide the penultimate precursor to glycine and therefore may represent a good target for down-regulation in addition to the role of glycine in the biosynthesis of purine nucleotides. Interestingly, these combinations of nucleotide and amino acid reactions, as shown, have not appeared in the double-reaction analysis for CHIKV and DENV. Potentially, we are seeing the effects of trade-off between a the increased demand of glycine and nucleotides in the ZIKV optimised system, that is not present in the CHIKV and DENV optimised ones (due to the respective virus biomass compositions). This particular trade off between serine/cysteine/glycine and nucleotide biosynthesis, as shown through the prediction of serine-associated and a purine-associated reactions, therefore goes some way in explaining why this effect is not observed in the other viruses. To put it simply, ZIKV requires a higher amount of glycine (compared to the host and the other viruses) that is utilised for *de novo* nucleotide biosynthesis and therefore presents a ZIKV-unique antiviral perturbation that effectively reduces the virus' ability to optimise in the constrained system.

Hypoxanthine, as part of the purine catabolic pathway, can be used to synthesise inosine monophosphate (IMP), the precursor to both guanine and adenosine-based RNA nucleotides. By upregulating the reactions that consume hypoxanthine a limitation is created on the amount that can be utilised for RNA purine nucleotide anabolism. The upregulation of hypoxanthine to inosine and deoxyinosine (as well as direct transport out of the cell) restricts purine salvage pathways and thus creates a network where IMP cannot be synthesised entirely in this manner. Combined with the downregulation of serine producing reactions, not only limiting a biomass component itself but also results in a limit on the amount of glycine available, which when combined with a disruption of the purine catabolic pathway, causes a decrease in overall virus biomass production.

As with CHIKV and DENV, reaction pairs that involve only amino acids are predicted as antiviral targets. In the case of ZIKV, the primary amino acids highlighted by this double reaction enforcement involve the amino acids arginine and proline (Fig. 4.13), both of which make up proportionally less of the ZIKV biomass that the host. This is reflected in the fact that the reactions identified all involve upregulation of arginine and proline producing pathways. More specifically, we

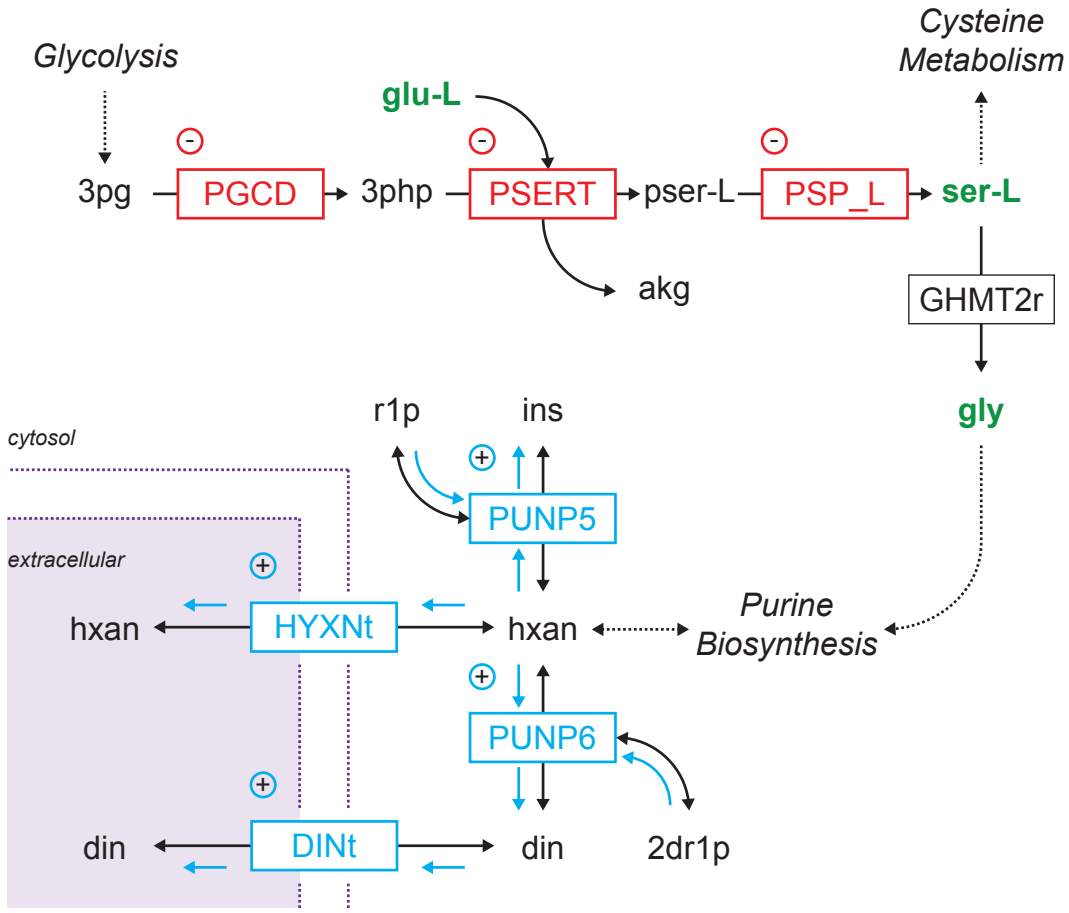


Figure 4.12: Zika virus double-reaction enforcements associated with serine-glycine metabolism and purine catabolism. Synergistic reaction pairs are formed between any (■) and (■) reactions, when simultaneously enforced. Reactions can be upregulated (+) or downregulated (-). Black arrows (➔) denote reaction directionality, and coloured arrows indicated the directionality of primary (➔) and secondary (➔) reaction pair members if reaction is bidirectional. Virus-associated biomass precursors are highlighted in green. Model compartment boundaries are indicated by dashed line (- - -). Reaction and metabolite abbreviations are expanded in the original model publication (Bordbar et al. 2010).

identify reaction pairs that are found in the urea cycle. Combinations of cytosolic arginase (ARGN) with either the cytosol-to-mitochondrial arginine transport reaction (ARGtm), mitochondrial arginase (ARGNm) or mitochondrial-to-cytosolic urea transport reaction (UREAtm) result in decrease ZIKV optima. In essence, these combinations cause a reduction in the availability of cytosolic arginine (the virus can only use amino acids that are cytosolic) and increase the availability of both cytosolic and mitochondrial ornithine (ORN), an important non-proteogenic amino acid used in proline biosynthesis. This represents a case where instead of limiting amino acid precursors, their overproduction causes a detrimental effect on virus production. In part, this is due to the limited metabolic resources available to the network, therefore by forcing the over-production of amino acids that form a lesser component of the viral biomass, the metabolic resources are consumed as apposed to being available for biosynthesis of other, more crucial biomass precursors.

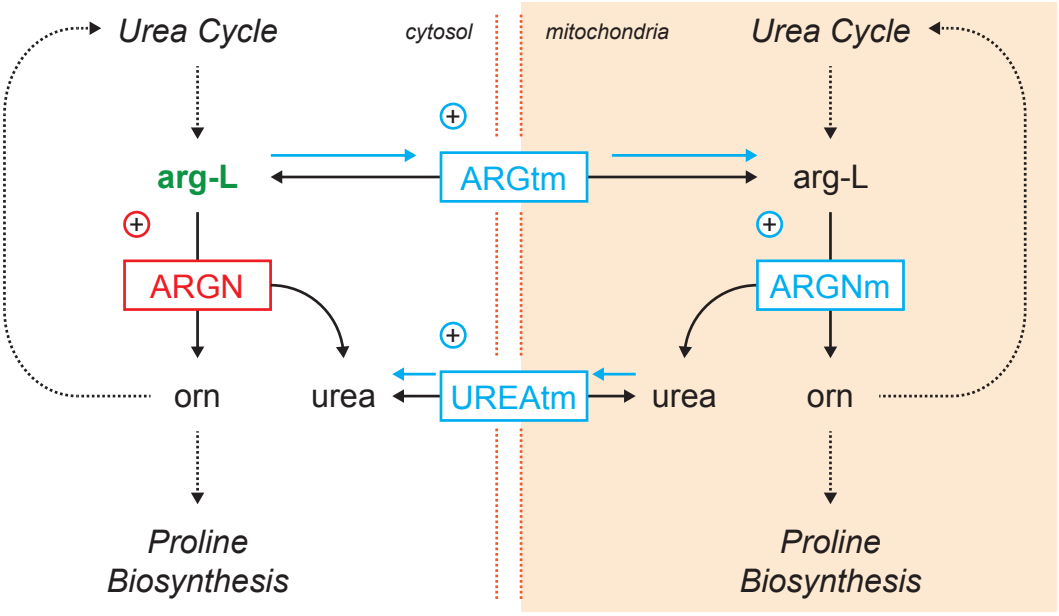


Figure 4.13: Zika virus double-reaction enforcements associated with arginine and proline metabolism. Reaction pairs are formed between any primary (■) and secondary (■) reactions, when simultaneously enforced. Reactions can be upregulated (+) or downregulated (-). Black arrows (➔) denote reaction directionality, and coloured arrows indicated the directionality of primary (➔) and secondary (➔) reaction pair members if reaction is bidirectional. Virus-associated biomass precursors are highlighted in green. Model compartment boundaries are indicated by dashed line (- - -). Reaction and metabolite abbreviations are expanded in the original model publication (Bordbar et al. 2010).

Summary. Double-reaction host-derived enforcements highlight a number of potentially novel antiviral targets for CHIKV, DENV and ZIKV. Broadly, these antiviral targets can be classified as amino acid or nucleotide targeting, based upon the metabolic pathways they fall into. CHIKV and DENV are susceptible to the most amino acid associated reactions, targeted towards the amino acids that contribute proportionally more to their overall biomass. ZIKV, whilst affected by amino acid targeting reactions, also is predict to be affected by nucleotide altering reaction pairs. Regardless of the types of reactions predicted to be effective in a pair, it is the case that pairs of up- and down-regulated reactions are the most effective, and yielded the greatest synergistic effect, of any host-derived enforcement pair. For many of these reactions, it was possible to identify existing inhibitory compounds from the DrugBank database (Wishart et al. 2008) that can potentially be experimentally explored as antiviral targets. Full tables of double-reaction enforcements targets for each virus, along with drug targets, are presented in Appendix F.

4.4 DISCUSSION

Here we present a novel computational methodology for predicting multiple control points (reactions) in an "infected" host-virus metabolic model. Building upon the previous chapter, where potential antiviral network perturbations were considered exhaustively in a single-reaction manner, we simulate the host-derived enforcement of reaction pairs in the model for all unique reaction combinations. This resulted in the simulation of 5757921 reaction pairs for each virus, totalling to 17273763 overall.

Compared to the single-reaction results, the double-reaction knockouts result in a greater number of virus optima reducing reactions. Due to the high number of completely reducing targets, an in-text table has not been included in this section (see Appendix F). However, as with the single-reaction knockout analysis, all targets are accompanied (if applicable) with a potential drug candidate(s) for the reaction(s). Similar to the single-reaction knockout results, the effect against the virus production simultaneously effects the host viability. Therefore, the single-reaction enforcement analysis was repeated in this double-reaction manner.

The double-reaction enforcements result in a larger number of potential antiviral reaction-combinations. Some of the reactions highlighted in this analysis as being

effective as a potential antiviral were up- or down-regulated. Whilst upregulated reactions may indeed provide some antiviral effect, they are potentially more difficult to achieve than a downregulated [inhibited] reaction, mainly due to a lack of viable drugs / compounds that achieve upregulation. It is most likely that the enzyme/gene itself would need to be upregulated in order to achieve this effect. We also acknowledge that whilst reactions that are labelled as compartment transporters (i.e. cytosol to mitochondria), such as hypoxanthine transport (HYXNt) shown in Fig. 4.12, may represent viable targets, they also typically lack identified drugs/compounds (in a similar manner to the upregulated reactions). This is due to a lack of information regarding the exact protein complex that is responsible for the transport, or due to the transportation being via a process such as diffusion.

Where possible, drug candidates have been identified and recorded for the reactions. However, some of these drugs are small molecules (i.e. NADH, GMP) and as such may not be suitable for antiviral therapeutic treatment. In some cases, drug candidates were not available at all. For both of these instances, two possibilities can be considered:

1. Searching other databases (i.e. BRENDA) for other potential inhibitors
2. Using short interfering RNAs (siRNAs) to inhibit the enzyme (using the EC number as an identifier for the gene-protein linked to that reaction) and therefore act as the novel antiviral therapeutic.

Overall, ZIKV shows the broadest range of effects (in terms of the virus optima reduction) but also the largest reduction in any case. CHIKV and DENV are both predicted to be affected by the double-reaction enforcements also, but not to the same extent for ZIKV.

CHAPTER 5

Conclusion

The aim of this PhD project was to develop an *in silico* modelling approach, using applications of flux balance analysis, to explore the metabolic interactions between a human macrophage host and three viruses: Chikungunya, Dengue and Zika viruses. To accomplish this the biomass composition, in terms of metabolic resources and energy requirements, had to be established for each virus being modelled. Additionally, these representations of the virus biomass (the biomass objective function) had to be integrated into a suitable host model, in this case a human macrophage metabolic network. Once these had been established and created, flux balance analysis was used to predict the differences in a host-optimised and virus-optimised network, the latter serving as a proxy for complete virus control. This method was then further built upon to identify sets of reactions (both in terms of a single set or sets of reaction pairs) that can be perturbed in such a way that elicit an antiviral effect, whilst attempting to preserve the host viability. This section will contain the findings of this thesis, and will be summarised in a number of key areas. These will be: the differences in host- and virus-optimised metabolic systems, predictions of single- and double-reaction antiviral targets, the role of viral biomass composition in the effectiveness of antiviral targets, and the implications that this has on antiviral development. Finally, future directions for this project will be discussed.

5.1 UNDERSTANDING THE INTERACTIONS BETWEEN HOST AND VIRUS: A METABOLIC VIEWPOINT

During this PhD study, one of the main aims was to use *in silico* methodologies to predict and explore the interactions between host organism's metabolic networks and viruses that infect and utilise the associated metabolic resources. The interplay between viruses and hosts, in terms of metabolic perturbations and rearrangements, has been established and experimentally verified (Miyake-Stoner & O'Shea 2014, Rodrigues et al. 2013, Delgado et al. 2010, Sanchez & Lagunoff 2015, Maynard et al. 2010). However, whilst these metabolic interactions have been characterised for some viruses, there still remains a lack of understanding in terms of the specific reaction modulation that occurs during these metabolic interactions. These studies are also particularly resource intensive, requiring the selection of a suitable viral host, development of assays and in some cases the amendment or development of specific methodologies, all of which can require significant amounts of time to implement (Gautam et al. 2017, Geisbert & Jahrling 2004, Morse et al. 2012). In an effort to alleviate some of these hurdles, the development of *in silico* systems proved potentially useful and insightful (Aurich & Thiele 2016, Kerkhoven et al. 2014). To this end, this project expanded upon previous attempts to model the interactions between host organisms and viruses. Previous studies had explored the metabolic interactions between bacteria and phages (Birch et al. 2012, Jain & Srivastava 2009), and the interactions between human hosts and bacterial pathogens (Bordbar et al. 2010), but these same techniques had not yet been applied to a human host and viral pathogen.

In chapter 2, a method for the description of a virus as a metabolic component of a host metabolic network was developed and used to integrate three viruses - Chikungunya, Dengue and Zika viruses - into a human macrophage host metabolic network (Bordbar et al. 2010). The idea behind this method, that viruses can be considered as an extra metabolic 'demand' on a host, is the driving force behind the majority of the research proposed in this thesis (Maynard et al. 2010, Mahmoudabadi et al. 2017). These models were then used to predict the differences in a host- and virus-optimised metabolic network, an abstraction that is used as a proxy for complete host- and viral-control of the metabolic network. Whilst the latter may not represent an actual biological state - one could argue that a virus can never completely control a metabolic system as this would inadvertently kill the host - the differences between the two states can inform metabolic interactions

that observed outside of the *in silico* environment (Heinken & Thiele 2015).

Flux predictions, for the host macrophage metabolic network, were completed for host- and virus-optimised systems. In general, the differences between the host- and virus-optimal states centred around the increased diversion of glycolytic fluxes, in the virus-optimal state, to amino acid and nucleotide *de novo* biosynthetic pathways (§2.3.2). These diversions related to the increased biomass-associated demands for amino acids and RNA nucleotides of the viral biomass in comparison to the host. This diversion of glycolytic flux, and indeed diversion of flux towards these biosynthetic pathways, lines up with previous research and studies that have identified the differences in metabolic regulation between uninfected and virus-infected cells (El-Bacha et al. 2004, Delgado et al. 2010, Carinhas et al. 2017, Enav et al. 2014). In particular, it is interesting to look at the differences between the individual viruses themselves. Whilst CHIKV and DENV are predicted to optimised the metabolic network so that glycolytic intermediates drive both the synthesis of nucleotide precursors, but also amino acids (serine, glycine and cysteine) and pyruvate, in the ZIKV optimised system there is a complete dissociation of glycolysis from serine-biosynthesis, pyruvate synthesis and indeed the feeding of the citric acid cycle (§2.3.2 and §2.3.3). Instead, under the control of ZIKV, the citric acid cycle is predominantly fed via glutamine catabolism, an effect that has been noted in other virus infected cells, mainly by the Vaccinia virus (Fontaine et al. 2014).

These results provide an insight into the specific reaction alterations that may occur during the transition from uninfected to virus-infected, in terms of the viral-mediate rearrangement of metabolic resources. However, these results need to be further explored with *in vitro* metabolic flux analysis studies in order to further promote both the understanding of the metabolic network perturbations during infection, but also to inform the model design and further improve the accuracy of the *in silico* predictions. Currently, there is a lack of data regarding the characterisation of intracellular fluxes, which hampers the development of this technique (Blank 2017). Future research will require the characterisation of these fluxes, in order to fully realise the systems biology cycle and spur the development of more accurate and elaborate metabolic models. However, the research conducted in this thesis provides a potential foundation, in terms of techniques used and the types of analyses performed, to initiate such an endeavour.

5.2 MODEL PREDICTIONS BETWEEN HOST- AND VIRUS-OPTIMAL METABOLIC STATES MATCHES METABOLITE-BASED OBSERVATIONS FROM INFECTED CELLS

As previously discussed, the model predictions arise from a thought-experiment in which we compare fluxes from the host metabolic system optimised for either host or virus objective functions. While a full shift of host metabolism to supporting viral production is unlikely, this comparison can still provide insights into how metabolic fluxes in a host might shift with subsequent viral infection. To see if the model predictions match with biological observations, a comparison was attempted between the general flux results, presented in chapter 2, with experimental data collected from controlled virus infection experiments involving the three viruses (CHIKV, DENV and ZIKV). Unfortunately, exploration of the literature failed to yield any studies that have directly measured metabolic flux changes across the course of a viral infection, for these viruses. There were, however, few datasets that considered changes in the cell medium or the serum upon infection, and subsequently found notable overlaps with this data and the flux predictions from this thesis project. For example, the model predictions for the upregulation of glycine, serine and threonine metabolic sub-processes (§2.3.2) in the CHIKV- and DENV-optimised models matches with previous metabolic studies of the serum of CHIKV and DENV infected humans (Shrinet et al. 2016).

There was also an attempt to compare the predictions with gene expression data from several infection experiments, presenting expression levels before and after infection. Unfortunately, none of these studies was conducted on the modelled host, the human macrophage cell, but instead used other human cell lines, and as a result there was no strong overall correlation between expression changes in metabolic genes and model-based flux changes (in line with the previously observed lack of correlation between enzyme expression and metabolic flux changes (Guo & Feng 2016)).

5.3 CREATING A SYSTEMS BIOLOGY PLATFORM FOR VIRAL METABOLIC MODELLING

Currently, multiple methods and computational applications exist for constraint-based metabolic modelling (Chindelevitch et al. 2014, Hoppe et al. 2011, Orth et al.

2010), although the most common application is the COBRA toolbox and associated code-specific extensions (Ebrahim et al. 2013). Whilst this toolbox is able to facilitate the creation, curation and analysis of metabolic models, it is unable to construct organism specific mathematical constructs (such as biomass objective functions). Therefore, part of this thesis project aimed to create an additional software layer for the COBRA toolbox, that would specifically be built for the creation and analysis of virus-infected metabolic models; this in-house software is called ViraNet, and its use and code is detailed in §2.2.7 and Appendix A.5, respectively. The full python-based program is also available through the OSS Lab Research Group computational resources website (http://osslab.lifesci.warwick.ac.uk/?pid=resources). The goal of creating this automated software, which requires minimal user input to create a host-virus integrated metabolic model, was to establish a systems biology platform that could spur further research into the metabolic analysis of host-virus interactions, as has previously been done for other combinations of organisms and indeed other systems biology based endeavours (Heinken & Thiele 2015, Munger et al. 2008, Kitano 2002).

5.4 PREDICTING ANTIVIRAL TARGETS: ISSUES OF METABOLIC ENTANGLEMENT

The overlap between a host and a virus, in terms of their composition and metabolic demands, are substantial: both organisms utilise a variety of amino acids and nucleotides to function, which therefore presents an issue when it comes to metabolite-based antiviral design (Maynard et al. 2010). As discussed throughout chapters 2-4, the overlap of biomass precursor metabolites between the human macrophage host and CHIKV, DENV, and ZIKV biomass objective functions, presented an issue in the prediction of effective antiviral targets. Initially, this project began with exploring the effect of knocking out a single reaction (by reducing the flux bounds of that reaction to zero, effectively removing it from the metabolic network) on the ability of the virus to optimise (§3.3.1). The results appeared promising; a number of reactions under the knockout constraint would render the virus (via the associated objective function reaction) unable to optimise within this constrained metabolic network. However, subsequent experiments predicted that the host biomass objective function was also unable to be optimised for, and as a result the targets also reduced host viability. These results highlighted the key point discussed earlier; because the host and virus overlap in terms of biomass precursors,

complete inhibition of an associated reaction (for those precursors) will ultimately end up affecting both the host and the virus (Ikeda & Kato 2007, Miyake-Stoner & O'Shea 2014). This trail of thought was what spurred the development of the host-derived enforcement reaction analysis (§3.2 and §3.3.2).

In essence, the host-derived enforcement analysis looked at how reactions could be partially perturbed (as opposed to entirely knocked-out) through up- or down-regulation, in order to decrease the predicted virus optima whilst simultaneously ensuring that the host optima did not deviate from the original prediction. This maintenance, or enforcement, of host viability enables the prediction of a set of antiviral targets that in theory maximise perturbation of the virus whilst minimising damage to the host. This premise contrasts with current metabolism-orientated [RNA] antiviral targets, most of which exclusively target *de novo* nucleotide biosynthesis (Leyssen et al. 2008, Khan et al. 2011, Jain et al. 2001, Wang et al. 2011, Chung 2015). Whilst these antivirals have been experimentally validated and shown to reduce the overall titre of infectious virus released from infected cells, they are also attributed to an increase in cell death relative to non-infected cells. This trade-off was found in chapter 3, whereby single-reaction knockouts in nucleotide biosynthetic pathways disabled the ability of the virus, but also the host, to optimise and use the metabolic network. Chapter 3 resolved this issue of host damage through the use of host-derived enforcements, and predicted small reductions in virus optima (between 2 and 15%). Whilst this proved promising, in terms of creating a method that confers host viability whilst perturbing the virus, the results can be criticised for only producing negligible effects against the viruses. Thus, in an attempt to further explore and expand upon this method, double-reactions were considered under the constraint of host derived enforcements (§4.3.1 - 4.3.4) for antiviral against CHIKV, DENV and ZIKV.

The double-reactions proved to be more successful, in terms of (a) the number of antiviral targets they predicted were effective and (b) the reduction of virus optima associated with those reaction pairings. What is particularly interesting, is whilst single-reaction antiviral targets were mainly predicted to affect the nucleotide biosynthetic pathways, the double-reaction pathways (in the cases of CHIKV and DENV) predominantly affected the amino acid biosynthetic pathways. In the case of ZIKV, both amino acid and nucleotide pathway targets yielded effective results. What is interesting is why this result has come about, particularly in the light of metabolic network topology.

The *de novo* nucleotide biosynthetic pathways, specifically purine and pyrimidine pathways, are non-branched and are a series of single input and single outputs, until the terminal branching event which results in the production of uracil-cytidine or adenosine-guanine nucleotides, respectively. Therefore, single-reaction perturbations are able to easily affect these synthesis pathways, as only one point of this network has to be perturbed in order to elicit a downstream effect. Vice versa, in the case of amino acids the *de novo* synthesis pathways are highly branched, with many interconnecting input and output reactions and multiple points of redundancy (several pathways to and from compounds) . This creates a situation where a single reaction is unable to perturb the amino acid biosynthesis enough to elicit an antiviral effect, but two reactions simultaneously perturbed are able to. This also helps explain why double-reactions are unable to produce a more substantial effect in the nucleotide biosynthetic pathways, as the single-reaction-chain perturbations affect all downstream reactions with equal effect. The largest issue faced, therefore, is how exactly to select pairs of reactions to yield the antiviral effect, the main result of which this thesis presents.

5.5 FUTURE DIRECTIONS

This thesis concludes with a set of reaction predictions, both for single- and double-reactions, that are effective against CHIKV, DENV and ZIKV. The most obvious future direction from this project is the experimental validation of the antiviral targets predicted in this project. At the time of writing, two collaborators are currently developing and implementing research projects that will experimentally validate the predicted targets, using drug compounds identified from the DrugBank database (Wishart et al. 2008) or through the use of small interfering RNAs (siRNAs), which specifically target the enzyme that catalyses the associated antiviral reaction target. These external collaborators, the Defence Science and Technology Laboratories (DSTL, UK) and University of Western Australia (UWA, Australia) will be initially focusing on CHIKV and ZIKV, respectively. The outputs of this thesis have inspired and informed these research projects, which upon completion will be further expanded to include DENV and other viruses that have as of yet not been computationally modelled.

This thesis also raises questions surrounding the influence of host-virus biomass composition mismatch as a potential avenue for further research. Throughout this thesis, the influence that stoichiometry has on antiviral targets, and indeed

flux predicts, has been clear: differing requirements for the amount of biomass precursors causes appropriate changes in the metabolic network. The interplay between the different precursors (nucleotides and amino acids) is therefore a logical next step, in order to determine how this may affect both the observed infection and antiviral predictions. Current literature and experimental evidence suggests that there are downstream effects of different virus compositions during infection, and that these differences play a role in how the virus interacts with the host cell (Mahmoudabadi et al. 2017). There are also observable differences in the amino acid and nucleotide levels within a cell (Dittmar et al. 2006, Martini et al. 2004, Bergström et al. 1974), and experiments have been conducted that show the impact this has on protein production (Raiford et al. 2008, Kaleta et al. 2013, Akashi & Gojobori 2002) but also that viruses can and have evolved to adapt to the host intracellular metabolic resources (Taubenberger & Kash 2010). Understanding this interplay between free metabolite resource concentrations and virus production and adaptation will aid in the discovery of effective antiviral targets.

5.6 FINAL CONCLUSIONS

Ultimately, this thesis concludes that the stoichiometric differences between host and virus drive the magnitude of the antiviral effect, predicted by the host-derived enforcements, for CHIKV, DENV and ZIKV. Nucleotide targets provide a more generic (in terms of the viruses they effect) antiviral effect that is successful against CHIKV, DENV and ZIKV. The amino acid targets, under the condition of double-reaction constraints, appear to offer more specificity in terms of their antiviral potential against different viruses. In the case of CHIKV, DENV and ZIKV, many of the reactions predicted to be effective against CHIKV were also effective against DENV. However, ZIKV had a number of reaction pairings that were not present, or had a smaller effect, in the other two viruses. This mainly revolved around the linkage between amino acids and nucleotides, specifically in the case of glycine (of which ZIKV has a substantially higher biomass demand than CHIKV or DENV) which is utilised as a nucleotide precursor. These biomass compositional differences seem to be what is driving the antiviral effects predicted, where trade-offs between host- and virus-requirements define the categories and types of antiviral effects that are observed.

Bibliography

Adams, M. J., Lefkowitz, E. J., King, A. M. Q. & Carstens, E. B. (2013), 'Recently agreed changes to the International Code of Virus Classification and Nomenclature.', *Arch. Virol.* **158**(12), 2633–2639.

Aderem, A., Adkins, J. N., Ansong, C., Galagan, J., Kaiser, S., Korth, M. J., Law, G. L., McDermott, J. G., Proll, S. C., Rosenberger, C., Schoolnik, G. & Katze, M. G. (2011), 'A systems biology approach to infectious disease research: innovating the pathogen-host research paradigm.', *MBio* **2**(1), e00325–10.

Akashi, H. & Gojobori, T. (2002), 'Metabolic efficiency and amino acid composition in the proteomes of Escherichia coli and Bacillus subtilis.', *Proceedings of the National Academy of Sciences* **99**(6), 3695–3700.

Almaas, E., Oltvai, Z. N. & Barabási, A.-L. (2005), 'The activity reaction core and plasticity of metabolic networks.', *PLoS Comput Biol* **1**(7), e68.

Antoniewicz, M. R. (2015), 'Methods and advances in metabolic flux analysis: a mini-review.', *J. Ind. Microbiol. Biotechnol.* **42**(3), 317–325.

Aurich, M. K. & Thiele, I. (2016), 'Computational Modeling of Human Metabolism and Its Application to Systems Biomedicine.', *Methods Mol. Biol.* **1386**(Chapter 12), 253–281.

Balsitis, S. J., Coloma, J., Castro, G., Alava, A., Flores, D., McKerrow, J. H., Beatty, P. R. & Harris, E. (2009), 'Tropism of dengue virus in mice and humans defined by viral nonstructural protein 3-specific immunostaining.', *Am. J. Trop. Med. Hyg.* **80**(3), 416–424.

Barsoum, N., Uatrongjit, S. & Vasant, P. (2008), Using ILOG OPL-CPLEX and ILOG Optimization Decision Manager (ODM) to Develop Better Models, *in* 'INTERNATIONAL CONFERENCE ON POWER CONTROL AND OPTIMIZATION: Innovation in Power Control for Optimal Industry', AIP, pp. 3–3.

Beard, D. A., Babson, E., Curtis, E. & Qian, H. (2004), 'Thermodynamic constraints for biochemical networks.', *J. Theor. Biol.* **228**(3), 327–333.

Bergström, J., Fürst, P., Norée, L. O. & Vinnars, E. (1974), 'Intracellular free amino acid concentration in human muscle tissue.', *Journal of Applied Physiology* **36**(6), 693–697.

Birch, E. W., Ruggero, N. A. & Covert, M. W. (2012), 'Determining host metabolic limitations on viral replication via integrated modeling and experimental perturbation.', *PLoS Comput Biol* **8**(10), e1002746.

Birch, E. W., Udell, M. & Covert, M. W. (2014), 'Incorporation of flexible objectives and time-linked simulation with flux balance analysis.', *J. Theor. Biol.* **345**, 12–21.

Blank, L. M. (2017), 'Let's talk about flux or the importance of (intracellular) reaction rates.', *Microb Biotechnol* **10**(1), 28–30.

Bonds, M. H., Dobson, A. P. & Keenan, D. C. (2012), 'Disease ecology, biodiversity, and the latitudinal gradient in income.', *PLoS Biol.* **10**(12), e1001456.

Bonomo, J. & Gill, R. T. (2005), 'Amino acid content of recombinant proteins influences the metabolic burden response.', *Biotechnol. Bioeng.* **90**(1), 116–126.

Bordbar, A., Lewis, N. E., Schellenberger, J., Palsson, B. Ø. & Jamshidi, N. (2010), 'Insight into human alveolar macrophage and M. tuberculosis interactions via metabolic reconstructions.', *Molecular Systems Biology* **6**, 422.

Bordbar, A., McCloskey, D., Zielinski, D. C., Sonnenschein, N., Jamshidi, N. & Palsson, B. Ø. (2015), 'Personalized Whole-Cell Kinetic Models of Metabolism for Discovery in Genomics and Pharmacodynamics.', *Cell Syst* **1**(4), 283–292.

Broyden, C. G. (1994), Linear Equations in Optimisation, *in* 'Algorithms for Continuous Optimization', Springer Netherlands, Dordrecht, pp. 25–35.

Carinhas, N., Koshkin, A., Pais, D. A. M., Alves, P. M. & Teixeira, A. P. (2017), '(13) C-metabolic flux analysis of human adenovirus infection: Implications for viral vector production.', *Biotechnol. Bioeng.* **114**(1), 195–207.

Cecchine, G. & Moore, M. (2006), 'Infectious Disease and National Security: Strategic Information Needs', *Edition: 1* .

Chambers, J. W., Maguire, T. G. & Alwine, J. C. (2010), 'Glutamine metabolism is essential for human cytomegalovirus infection.', *Journal of Virology* **84**(4), 1867–1873.

Chang, R. L., Xie, L., Xie, L., Bourne, P. E. & Palsson, B. Ø. (2010), 'Drug off-target effects predicted using structural analysis in the context of a metabolic network model.', *PLoS Comput Biol* **6**(9), e1000938.

Chindelevitch, L., Trigg, J., Regev, A. & Berger, B. (2014), 'An exact arithmetic toolbox for a consistent and reproducible structural analysis of metabolic network models.', *Nature Communications* **5**, 4893.

Chowdhury, A. & Maranas, C. D. (2015), 'Personalized Kinetic Models for Predictive Healthcare.', *Cell Syst* **1**(4), 250–251.

Chung, D. (2015), 'The Establishment of an Antiviral State by Pyrimidine Synthesis Inhibitor is Cell Type-Specific', *Journal of Antimicrobial Agents* **01**(01).

Clercq, E. D. (2004), 'Antivirals and antiviral strategies', *Nat. Rev. Microbiol.* **2**(9), 704–720.

Colijn, C., Brandes, A., Zucker, J., Lun, D. S., Weiner, B., Farhat, M. R., Cheng, T.-Y., Moody, D. B., Murray, M. & Galagan, J. E. (2009), 'Interpreting Expression Data with Metabolic Flux Models: Predicting Mycobacterium tuberculosis Mycolic Acid Production', *PLoS Comput Biol* **5**(8), e1000489.

Coscia, V., Fermo, L. & Bellomo, N. (2011), 'On the mathematical theory of living systems II: The interplay between mathematics and system biology', *Computers & Mathematics with Applications* **62**(10), 3902–3911.

Dantzig, G. B. & Thapa, M. N. (1997), 'Linear programming: 1: Introduction (Springer series in operations research and financial engineering)'.

DeBerardinis, R. J., Lum, J. J., Hatzivassiliou, G. & Thompson, C. B. (2008), 'The biology of cancer: metabolic reprogramming fuels cell growth and proliferation.', *Cell Metabolism* **7**(1), 11–20.

DeBerardinis, R. J., Mancuso, A., Daikhin, E., Nissim, I., Yudkoff, M., Wehrli, S. & Thompson, C. B. (2007), 'Beyond aerobic glycolysis: transformed cells can engage in glutamine metabolism that exceeds the requirement for protein and nucleotide synthesis.', *Proc. Natl. Acad. Sci. U.S.A.* **104**(49), 19345–19350.

Delgado, T., Carroll, P. A., Punjabi, A. S., Margineantu, D., Hockenbery, D. M. & Lagunoff, M. (2010), 'Induction of the Warburg effect by Kaposi's sarcoma herpesvirus is required for the maintenance of latently infected endothelial cells.', *Proc. Natl. Acad. Sci. U.S.A.* **107**(23), 10696–10701.

Deng, L., Shoji, I., Ogawa, W., Kaneda, S., Soga, T., Jiang, D.-p., Ide, Y.-H. & Hotta, H. (2011), 'Hepatitis C virus infection promotes hepatic gluconeogenesis through an NS5A-mediated, FoxO1-dependent pathway.', *Journal of Virology* **85**(17), 8556–8568.

Desouki, A. A., Jarre, F., Gelius-Dietrich, G. & Lercher, M. J. (2015), 'CycleFreeFlux: efficient removal of thermodynamically infeasible loops from flux distributions', *Bioinformatics* **31**(13), 2159–2165.

Dhurjati, P. & Mahadevan, R. (2008), 'Systems Biology: The synergistic interplay between biology and mathematics', *The Canadian Journal of Chemical Engineering* **86**(2), 127–141.

Dittmar, K. A., Goodenbour, J. M. & Pan, T. (2006), 'Tissue-specific differences in human transfer RNA expression.', *PLoS Genet.* **2**(12), e221.

Duarte, N. C., Becker, S. A., Jamshidi, N., Thiele, I., Mo, M. L., Vo, T. D., Srivas, R. & Palsson, B. Ø. (2007), 'Global reconstruction of the human metabolic network based on genomic and bibliomic data.', *Proceedings of the National Academy of Sciences* **104**(6), 1777–1782.

Dunn, R. R., Davies, T. J., Harris, N. C. & Gavin, M. C. (2010), 'Global drivers of human pathogen richness and prevalence.', *Proc. Biol. Sci.* **277**(1694), 2587–2595.

Durmuş, S., Çakır, T., Özgür, A. & Guthke, R. (2015), 'A review on computational systems biology of pathogen-host interactions.', *Front. Microbiol* **6**(S2), 235.

Ebrahim, A., Lerman, J. A., Palsson, B. Ø. & Hyduke, D. R. (2013), 'COBRApy: COnstraints-Based Reconstruction and Analysis for Python.', *BMC Syst Biol* **7**(1), 74.

Eisenreich, W., Heesemann, J., Rudel, T. & Goebel, W. (2013), 'Metabolic host responses to infection by intracellular bacterial pathogens.', *Front Cell Infect Microbiol* **3**, 24.

El-Bacha, T., Menezes, M. M. T., Azevedo e Silva, M. C., Sola-Penna, M. & Da Poian, A. T. (2004), 'Mayaro virus infection alters glucose metabolism in cultured cells through activation of the enzyme 6-phosphofructo 1-kinase.', *Mol. Cell. Biochem.* **266**(1-2), 191–198.

Enav, H., Mandel-Gutfreund, Y. & Béjà, O. (2014), 'Comparative metagenomic analyses reveal viral-induced shifts of host metabolism towards nucleotide biosynthesis.', *Microbiome* **2**(1), 9.

Endy, D. & Yin, J. (2000), 'Toward antiviral strategies that resist viral escape.', *Antimicrob. Agents Chemother.* **44**(4), 1097–1099.

Eubank, S. (2005), 'Network based models of infectious disease spread.', *Jpn. J. Infect. Dis.* **58**(6), S9–13.

Fauci, A. S., Touchette, N. A. & Folkers, G. K. (2005), 'Emerging infectious diseases: a 10-year perspective from the National Institute of Allergy and Infectious Diseases.', *Emerging Infect. Dis.* **11**(4), 519–525.

Feist, A. M. & Palsson, B. Ø. (2010), 'The biomass objective function', *Current Opinion in Microbiology* .

Fell, D. A. (2010), 'Evolution of central carbon metabolism.', *Molecular Cell* **39**(5), 663–664.

Figueiredo, M. L. G. d. & Figueiredo, L. T. M. (2014), 'Emerging alphaviruses in the Americas: Chikungunya and Mayaro.', *Rev. Soc. Bras. Med. Trop.* **47**(6), 677–683.

Fischl, W. & Bartenschlager, R. (2011), 'Exploitation of cellular pathways by Dengue virus.', *Current Opinion in Microbiology* **14**(4), 470–475.

Fontaine, K. A., Camarda, R. & Lagunoff, M. (2014), 'Vaccinia virus requires glutamine but not glucose for efficient replication.', *Journal of Virology* **88**(8), 4366–4374.

Fontaine, K. A., Sanchez, E. L., Camarda, R. & Lagunoff, M. (2015), 'Dengue virus induces and requires glycolysis for optimal replication.', *Journal of Virology* **89**(4), 2358–2366.

Fox, J. M., Long, F., Edeling, M. A., Lin, H., van Duijl-Richter, M. K. S., Fong, R. H., Kahle, K. M., Smit, J. M., Jin, J., Simmons, G., Doranz, B. J., Crowe, J. E., Fremont, D. H., Rossmann, M. G. & Diamond, M. S. (2015), 'Broadly Neutralizing Alphavirus Antibodies Bind an Epitope on E2 and Inhibit Entry and Egress.', *Cell* **163**(5), 1095–1107.

Galván-Peña, S. & O'Neill, L. A. J. (2014), 'Metabolic reprograming in macrophage polarization.', *Front Immunol* **5**(1), 420.

Garmashova, N., Gorchakov, R., Volkova, E., Paessler, S., Frolova, E. & Frolov, I. (2007), 'The Old World and New World alphaviruses use different virus-specific proteins for induction of transcriptional shutoff.', *Journal of Virology* **81**(5), 2472–2484.

Gautam, R., Mishra, S., Milhotra, A., Nagpal, R., Mohan, M., Singhal, A. & Kumari, P. (2017), 'Challenges with Mosquito-borne Viral Diseases: Outbreak of the Monsters.', *Curr Top Med Chem* **17**(19), 2199–2214.

Geer, L. Y., Marchler-Bauer, A., Geer, R. C., Han, L., He, J., He, S., Liu, C., Shi, W. & Bryant, S. H. (2010), 'The NCBI BioSystems database.', *Nucleic Acids Research* **38**(Database issue), D492–6.

Geisbert, T. W. & Jahrling, P. B. (2004), 'Exotic emerging viral diseases: progress and challenges.', *Nat. Med.* **10**(12 Suppl), S110–21.

Gilbert, S. F., Sapp, J. & Tauber, A. I. (2012), 'A symbiotic view of life: we have never been individuals.', *Q Rev Biol* **87**(4), 325–341.

Gille, C., Bölling, C., Hoppe, A., Bulik, S., Hoffmann, S., Hübner, K., Karlstädt, A., Ganeshan, R., König, M., Rother, K., Weidlich, M., Behre, J. & Holzhütter, H.-G. (2010), 'HepatoNet1: a comprehensive metabolic reconstruction of the human hepatocyte for the analysis of liver physiology.', *Molecular Systems Biology* **6**, 411.

Gollins, S. W. & Porterfield, J. S. (1985), 'Flavivirus infection enhancement in macrophages: an electron microscopic study of viral cellular entry.', *J. Gen. Virol.* **66 (Pt 9)**(9), 1969–1982.

Goodwin, C. M., Xu, S. & Munger, J. (2015), 'Stealing the keys to the kitchen: Viral manipulation of the host cell metabolic network', *Trends in Microbiology* .

Grantham, R. (1974), 'Amino acid difference formula to help explain protein evolution.', *Science* **185**(4154), 862–864.

Grutman, M. I. & Orgel, M. I. (1970), 'Nucleic acid content in cultures of macrophages with antigen', *Biull Eksp Biol Med* **70**(7), 70–71.

Gubler, D. J. (2002), 'The global emergence/resurgence of arboviral diseases as public health problems.', *Arch. Med. Res.* **33**(4), 330–342.

Guernier, V., Hochberg, M. E. & Guégan, J.-F. (2004), 'Ecology drives the worldwide distribution of human diseases.', *PLoS Biol.* **2**(6), e141.

Guimerà, R. & Nunes Amaral, L. A. (2005), 'Functional cartography of complex metabolic networks.', *Nature* **433**(7028), 895–900.

Guo, W. & Feng, X. (2016), 'OM-FBA: Integrate Transcriptomics Data with Flux Balance Analysis to Decipher the Cell Metabolism.', *PLoS ONE* **11**(4), e0154188.

Gutierrez, J. M. & Lewis, N. E. (2015), 'Optimizing eukaryotic cell hosts for protein production through systems biotechnology and genome-scale modeling.', *Biotechnol J* **10**(7), 939–949.

Hatzimanikatis, V., Li, C., Ionita, J. A., Henry, C. S., Jankowski, M. D. & Broadbelt, L. J. (2005), 'Exploring the diversity of complex metabolic networks.', *Bioinformatics* **21**(8), 1603–1609.

Hayden, F. G. (1996), 'Combination antiviral therapy for respiratory virus infections.', *Antiviral Research* **29**(1), 45–48.

Haynie, D. T. (2009), *Biological Thermodynamics*, 2 edn, Cambridge University Press, Cambridge.

Heinken, A. & Thiele, I. (2015), 'Systems biology of host-microbe metabolomics.', *Wiley Interdiscip Rev Syst Biol Med* **7**(4), 195–219.

Henry, C. S., DeJongh, M., Best, A. A., Frybarger, P. M., Linsay, B. & Stevens, R. L. (2010), 'High-throughput generation, optimization and analysis of genome-scale metabolic models.', *Nat. Biotechnol.* **28**(9), 977–982.

Hoffmann, H. H., Kunz, A. & Simon, V. A. (2011), Broad-spectrum antiviral that interferes with de novo pyrimidine biosynthesis, *in* 'Proceedings of the National Academy of Sciences'.

Hoppe, A., Hoffmann, S., Gerasch, A., Gille, C. & Holzhütter, H.-G. (2011), 'FASIMU: flexible software for flux-balance computation series in large metabolic networks.', *BMC Bioinformatics* **12**(1), 28.

Hoppe, A., Hoffmann, S. & Holzhütter, H.-G. (2007), 'Including metabolite concentrations into flux balance analysis: thermodynamic realizability as a constraint on flux distributions in metabolic networks.', *BMC Syst Biol* **1**, 23.

Howard, C. R. & Fletcher, N. F. (2012), 'Emerging virus diseases: can we ever expect the unexpected?', *Emerg Microbes Infect* **1**(12), e46–e46.

Hulo, C., de Castro, E., Masson, P., Bougueleret, L., Bairoch, A., Xenarios, I. & Le Mercier, P. (2011), 'ViralZone: a knowledge resource to understand virus diversity.', *Nucleic Acids Research* **39**(Database issue), D576–82.

Ibarra, R. U., Edwards, J. S. & Palsson, B. Ø. (2002), 'Escherichia coli K-12 undergoes adaptive evolution to achieve in silico predicted optimal growth.', *Nature* **420**(6912), 186–189.

Ikeda, M. & Kato, N. (2007), 'Modulation of host metabolism as a target of new antivirals.', *Adv. Drug Deliv. Rev.* **59**(12), 1277–1289.

Iyengar, U. & UK, V. (1985), 'Enumeration and structural assessment of peritoneal macrophages during progressive protein deficiency in rats', *Journal of Bioscience* **7**, 15–26.

Jahan, N., Maeda, K., Matsuoka, Y., Sugimoto, Y. & Kurata, H. (2016), 'Development of an accurate kinetic model for the central carbon metabolism of Escherichia coli.', *Microb. Cell Fact.* **15**(1), 112.

Jain, J., Almquist, S. J. & Shlyakhter, D. (2001), 'VX-497: A novel, selective IMPDH inhibitor and immunosuppressive agent', *Journal of ...* .

Jain, R. & Srivastava, R. (2009), 'Metabolic investigation of host/pathogen interaction using MS2-infected Escherichia coli.', *BMC Syst Biol* **3**, 121.

Jamshidi, N. & Raghunathan, A. (2015), 'Cell scale host-pathogen modeling: another branch in the evolution of constraint-based methods', *Front. Microbiol* **6**(299), 31522.

Jhuang, H.-J., Hsu, W.-H., Lin, K.-T., Hsu, S.-L., Wang, F.-S., Chou, C.-K., Lee, K.-H., Tsou, A.-P., Lai, J.-M., Yeh, S.-F. & Huang, C.-Y. F. (2015), 'Gluconeogenesis, lipogenesis, and HBV replication are commonly regulated by PGC-1α-dependent pathway.', *Oncotarget* **6**(10), 7788–7803.

Kaleta, C., Schäuble, S., Rinas, U. & Schuster, S. (2013), 'Metabolic costs of amino acid and protein production in Escherichia coli.', *Biotechnol J* **8**(9), 1105–1114.

Kanehisa, M., Furumichi, M., Tanabe, M., Sato, Y. & Morishima, K. (2017), 'KEGG: new perspectives on genomes, pathways, diseases and drugs.', *Nucleic Acids Research* **45**(D1), D353–D361.

Karupiah, G. & Harris, N. (1995), 'Inhibition of viral replication by nitric oxide and its reversal by ferrous sulfate and tricarboxylic acid cycle metabolites.', *J. Exp. Med.* **181**(6), 2171–2179.

Kauffman, K. J., Prakash, P. & Edwards, J. S. (2003), 'Advances in flux balance analysis.', *Curr. Opin. Biotechnol.* **14**(5), 491–496.

Kazeros, A., Harvey, B.-G., Carolan, B. J., Vanni, H., Krause, A. & Crystal, R. G. (2008), 'Overexpression of apoptotic cell removal receptor MERTK in alveolar macrophages of cigarette smokers.', *Am. J. Respir. Cell Mol. Biol.* **39**(6), 747–757.

Keeling, M. J. & Eames, K. T. D. (2005), 'Networks and epidemic models.', *Journal of The Royal Society Interface* **2**(4), 295–307.

Kerkhoven, E. J., Lahtvee, P.-J. & Nielsen, J. (2014), 'Applications of computational modeling in metabolic engineering of yeast', *FEMS Yeast Res* **341**, n/a–n/a.

Khan, M., Dhanwani, R., Patro, I. K., Rao, P. V. L. & Parida, M. M. (2011), 'Cellular IMPDH enzyme activity is a potential target for the inhibition of Chikungunya virus replication and virus induced apoptosis in cultured mammalian cells.', *Antiviral Research* **89**(1), 1–8.

King, Z. A., Dräger, A., Ebrahim, A., Sonnenschein, N., Lewis, N. E. & Palsson, B. Ø. (2015), 'Escher: A Web Application for Building, Sharing, and Embedding Data-Rich Visualizations of Biological Pathways.', *PLoS Comput Biol* **11**(8), e1004321.

Kitano, H. (2002), 'Systems Biology: A Brief Overview', *Science* **295**(5560), 1662–1664.

Kotte, O., Zaugg, J. B. & Heinemann, M. (2010), 'Bacterial adaptation through distributed sensing of metabolic fluxes.', *Molecular Systems Biology* **6**, 355.

Kotzamanis, K., Angulo, A. & Ghazal, P. (2015), 'Infection homeostasis: implications for therapeutic and immune programming of metabolism in controlling infection.', *Med. Microbiol. Immunol.* **204**(3), 395–407.

Kussow, C. M., Zhou, W., Gryte, D. M. & Hu, W.-S. (1995), 'Monitoring of mammalian cell growth and virus production process using on-line oxygen uptake rate measurement', *Enzyme and Microbial Technology* **17**(9), 779–783.

Law, G. L., Korth, M. J., Benecke, A. G. & Katze, M. G. (2013), 'Systems virology: host-directed approaches to viral pathogenesis and drug targeting.', *Nature Publishing Group* **11**(7), 455–466.

Le Novère, N., Finney, A., Hucka, M., Bhalla, U. S., Campagne, F., Collado-Vides, J., Crampin, E. J., Halstead, M., Klipp, E., Mendes, P., Nielsen, P., Sauro, H., Shapiro, B., Snoep, J. L., Spence, H. D. & Wanner, B. L. (2005), 'Minimum information requested in the annotation of biochemical models (MIRIAM).', *Nat. Biotechnol.* **23**(12), 1509–1515.

Lenski, R. E. & Travisano, M. (1994), 'Dynamics of adaptation and diversification: a 10,000-generation experiment with bacterial populations.', *Proceedings of the National Academy of Sciences* **91**(15), 6808–6814.

Lewis, N. E., Hixson, K. K., Conrad, T. M., Lerman, J. A., Charusanti, P., Polpitiya, A. D., Adkins, J. N., Schramm, G., Purvine, S. O., Lopez-Ferrer, D., Weitz, K. K., Eils, R., König, R., Smith, R. D. & Palsson, B. Ø. (2010), 'Omic data from evolved E. coli are consistent with computed optimal growth from genome-scale models.', *Molecular Systems Biology* **6**, 390–13.

Leyssen, P., De Clercq, E. & Neyts, J. (2008), 'Molecular strategies to inhibit the replication of RNA viruses.', *Antiviral Research* **78**(1), 9–25.

Littler, E. & Oberg, B. (2005), 'Achievements and challenges in antiviral drug discovery.', *Antivir. Chem. Chemother.* **16**(3), 155–168.

Liu, J. K., O'Brien, E. J., Lerman, J. A., Zengler, K., Palsson, B. Ø. & Feist, A. M. (2014), 'Reconstruction and modeling protein translocation and compartmentalization in Escherichia coli at the genome-scale.', *BMC Syst Biol* **8**(1), 110.

Lundström, J. O. (1999), 'Mosquito-borne viruses in western Europe: a review.', *J Vector Ecol* **24**(1), 1–39.

Luz, K. G. (2016), 'Dengue, Chikungunya and Zika Virus: Global emergence', *International Journal of Infectious Diseases* **45**, 39.

Lwande, O. W., Obanda, V., Bucht, G., Mosomtai, G., Otieno, V., Ahlm, C. & Evander, M. (2015), 'Global emergence of Alphaviruses that cause arthritis in humans.', *Infect Ecol Epidemiol* **5**, 29853.

Ma, H., Sorokin, A., Mazein, A., Selkov, A., Selkov, E., Demin, O. & Goryanin, I. (2007), 'The Edinburgh human metabolic network reconstruction and its functional analysis.', *Molecular Systems Biology* **3**, 135.

Mahadevan, R., Edwards, J. S. & Doyle III, F. J. (2002), 'Dynamic Flux Balance Analysis of Diauxic Growth in Escherichia coli', *Biophysical Journal* **83**(3), 1331–1340.

Mahadevan, R. & Schilling, C. H. (2003), 'The effects of alternate optimal solutions in constraint-based genome-scale metabolic models.', *Metabolic Engineering* **5**(4), 264–276.

Mahmoudabadi, G., Milo, R. & Phillips, R. (2017), 'Energetic cost of building a virus.', *Proc. Natl. Acad. Sci. U.S.A.* **114**(22), E4324–E4333.

Marquez, V. E., Lim, M. I., Treanor, S. P., Plowman, J., Priest, M. A., Markovac, A., Khan, M. S., Kaskar, B. & Driscoll, J. S. (1988), 'Cyclopentenylcytosine. A carbocyclic nucleoside with antitumor and antiviral properties.', *J. Med. Chem.* **31**(9), 1687–1694.

Martín-Jiménez, C. A., Salazar-Barreto, D., Barreto, G. E. & González, J. (2017), 'Genome-Scale Reconstruction of the Human Astrocyte Metabolic Network.', *Front Aging Neurosci* **9**, 23.

Martini, W. Z., Chinkes, D. L. & Wolfe, R. R. (2004), 'The intracellular free amino acid pool represents tracer precursor enrichment for calculation of protein synthesis in cultured fibroblasts and myocytes.', *J. Nutr.* **134**(6), 1546–1550.

Matsuoka, Y. & Shimizu, K. (2013), 'Catabolite regulation analysis of Escherichia coli for acetate overflow mechanism and co-consumption of multiple sugars based on systems biology approach using computer simulation.', *Journal of Biotechnology* **168**(2), 155–173.

Maynard, N. D., Gutschow, M. V., Birch, E. W. & Covert, M. W. (2010), 'The virus as metabolic engineer.', *Biotechnol J* **5**(7), 686–694.

McKinney, W. (2010), Data Structures for Statistical Computing in Python, *in* S. van der Walt & J. Millman, eds, 'Proceedings of the th Python in Science Conference', Proceedings of the th Python in Science Conference, pp. 51–56.

Meadows, A. L., Karnik, R., Lam, H., Forestell, S. & Snedecor, B. (2010), 'Application of dynamic flux balance analysis to an industrial Escherichia coli fermentation.', *Metabolic Engineering* **12**(2), 150–160.

Millman, K. J. & Aivazis, M. (2011), 'Python for Scientists and Engineers', *Computing in Science & Engineering* **13**(2), 9–12.

Miyake-Stoner, S. J. & O'Shea, C. C. (2014), 'Metabolism goes viral.', *Cell Metabolism* **19**(4), 549–550.

Mlakar, J., Korva, M., Tul, N., Popović, M., Poljšak-Prijatelj, M., Mraz, J., Kolenc, M., Resman Rus, K., Vesnaver Vipotnik, T., Fabjan Vodušek, V., Vizjak, A., Pižem, J., Petrovec, M. & Avšič Županc, T. (2016), 'Zika Virus Associated with Microcephaly.', *N. Engl. J. Med.* **374**(10), 951–958.

Molenaar, D., van Berlo, R., de Ridder, D. & Teusink, B. (2009), 'Shifts in growth strategies reflect tradeoffs in cellular economics.', *Molecular Systems Biology* **5**, 323.

Morse, S. S., Mazet, J. A. K., Woolhouse, M., Parrish, C. R., Carroll, D., Karesh, W. B., Zambrana-Torrelio, C., Lipkin, W. I. & Daszak, P. (2012), 'Prediction and prevention of the next pandemic zoonosis.', *Lancet* **380**(9857), 1956–1965.

Motamedian, E. (2015), 'A new algorithm to find all alternate optimal flux distributions of a metabolic network', *Computers & Chemical Engineering* **73**, 64–69.

Mukhopadhyay, S., Kuhn, R. J. & Rossmann, M. G. (2005), 'A structural perspective of the flavivirus life cycle.', *Nat. Rev. Microbiol.* **3**(1), 13–22.

Munger, J., Bennett, B. D., Parikh, A., Feng, X.-J., McArdle, J., Rabitz, H. A., Shenk, T. & Rabinowitz, J. D. (2008), 'Systems-level metabolic flux profiling identifies fatty acid synthesis as a target for antiviral therapy.', *Nat. Biotechnol.* **26**(10), 1179–1186.

Nering, E. D. & Tucker, A. W. (1992), 'Linear Programs & Related Problems: A Volume in the Computer Science and Scientific Computing Series'.

Newsholme, P., Curi, R., Gordon, S. & Newsholme, E. A. (1986), 'Metabolism of glucose, glutamine, long-chain fatty acids and ketone bodies by murine macrophages.', *Biochemical Journal* **239**(1), 121–125.

Newsholme, P., Curi, R., Pithon Curi, T. C., Murphy, C. J., Garcia, C. & Pires de Melo, M. (1999), 'Glutamine metabolism by lymphocytes, macrophages, and neutrophils: its importance in health and disease.', *J. Nutr. Biochem.* **10**(6), 316–324.

Nomaguchi, M., Fujita, M., Miyazaki, Y. & Adachi, A. (2012), 'Viral Tropism', *Front. Microbiol* **3**.

Orth, J. D., Thiele, I. & Palsson, B. Ø. (2010), 'What is flux balance analysis?', *Nat. Biotechnol.* **28**(3), 245–248.

Pál, C., Papp, B. & Lercher, M. J. (2005), 'Adaptive evolution of bacterial metabolic networks by horizontal gene transfer', *Nature Genetics* **37**(12), 1372–1375.

Pál, C., Papp, B., Lercher, M. J., Csermely, P., Oliver, S. G. & Hurst, L. D. (2006), 'Chance and necessity in the evolution of minimal metabolic networks.', *Nature* **440**(7084), 667–670.

Palomares, L. A., López, S. & RamÄśÌĄrez, O. T. (2004), 'Utilization of oxygen uptake rate to assess the role of glucose and glutamine in the metabolism of infected insect cell cultures', *Biochemical Engineering Journal* **19**(1), 87–93.

Papin, J. A., Price, N. D., Wiback, S. J. & Fell, D. A. (2003), 'Metabolic pathways in the post-genome era', *Trends in biochemical ...* .

Papin, J. A., Stelling, J., Price, N. D., Klamt, S., Schuster, S. & Palsson, B. Ø. (2004), 'Comparison of network-based pathway analysis methods.', *Trends Biotechnol.* **22**(8), 400–405.

Perelson, A. S. & Ribeiro, R. M. (2013), 'Modeling the within-host dynamics of HIV infection.', *BMC Biol.* **11**, 96.

Plata, G., Hsiao, T.-L., Olszewski, K. L., Llinás, M. & Vitkup, D. (2010), 'Reconstruction and flux-balance analysis of the Plasmodium falciparum metabolic network.', *Molecular Systems Biology* **6**, 408.

Poordad, F., Lawitz, E., Kowdley, K. V., Cohen, D. E., Podsadecki, T., Siggelkow, S., Heckaman, M., Larsen, L., Menon, R., Koev, G., Tripathi, R., Pilot-Matias, T. & Bernstein, B. (2013), 'Exploratory Study of Oral Combination Antiviral Therapy for Hepatitis C', *N. Engl. J. Med.* **368**(1), 45–53.

Price, N. D., Papin, J. A., Schilling, C. H. & Palsson, B. Ø. (2003), 'Genome-scale microbial in silico models: the constraints-based approach', *Trends Biotechnol.* **21**(4), 163–169.

Quek, L.-E., Dietmair, S., Krömer, J. O. & Nielsen, L. K. (2010), 'Metabolic flux analysis in mammalian cell culture', *Metabolic Engineering* **12**(2), 161–171.

Raiford, D. W., Heizer, E. M., Miller, R. V., Akashi, H., Raymer, M. L. & Krane, D. E. (2008), 'Do amino acid biosynthetic costs constrain protein evolution in Saccharomyces cerevisiae?', *J Mol Evol* **67**(6), 621–630.

Ravasz, E., Somera, A. L., Mongru, D. A., Oltvai, Z. N. & Barabási, A. L. (2002), 'Hierarchical organization of modularity in metabolic networks.', *Science* **297**(5586), 1551–1555.

Razonable, R. R. (2011), 'Antiviral Drugs for Viruses Other Than Human Immunodeficiency Virus', *Mayo Clinic Proceedings* **86**(10), 1009–1026.

Resat, H., Petzold, L. & Pettigrew, M. F. (2009), Kinetic Modeling of Biological Systems, *in* 'Computational Systems Biology', Humana Press, Totowa, NJ, pp. 311–335.

Ritter, J. B., Wahl, A. S., Freund, S., Genzel, Y. & Reichl, U. (2010), 'Metabolic effects of influenza virus infection in cultured animal cells: Intra- and extracellular metabolite profiling.', *BMC Syst Biol* **4**(1), 61.

Rodrigues, A. F., Formas-Oliveira, A. S., Bandeira, V. S., Alves, P. M., Hu, W. S. & Coroadinha, A. S. (2013), 'Metabolic pathways recruited in the production of a recombinant enveloped virus: mining targets for process and cell engineering.', *Metabolic Engineering* **20**, 131–145.

Rudge, T. J., Steiner, P. J., Phillips, A. & Haseloff, J. (2012), 'Computational Modeling of Synthetic Microbial Biofilms', *ACS Synth. Biol.* **1**(8), 345–352.

Ryu, J. Y., Kim, H. U. & Lee, S. Y. (2015), 'Reconstruction of genome-scale human metabolic models using omics data.', *Integr Biol (Camb)* **7**(8), 859–868.

Saa, P. A. & Nielsen, L. K. (2016), 'Construction of feasible and accurate kinetic models of metabolism: A Bayesian approach', *Nature Publishing Group* **6**(1), 57.

Sahoo, S., Franzson, L., Jonsson, J. J. & Thiele, I. (2012), 'A compendium of inborn errors of metabolism mapped onto the human metabolic network.', *Molecular BioSystems* **8**(10), 2545–2558.

Sanchez, E. L., Carroll, P. A., Thalhofer, A. B. & Lagunoff, M. (2015), 'Latent KSHV Infected Endothelial Cells Are Glutamine Addicted and Require Glutaminolysis for Survival.', *PLoS Pathog.* **11**(7), e1005052.

Sanchez, E. L. & Lagunoff, M. (2015), 'Viral activation of cellular metabolism.', *Virology* **479-480**, 609–618.

Sato, H., Watanabe, H., Ishii, T. & Bannai, S. (1987), 'Neutral amino acid transport in mouse peritoneal macrophages.', *J. Biol. Chem.* **262**(27), 13015–13019.

Schmidt, S., Sunyaev, S., Bork, P. & Dandekar, T. (2003), 'Metabolites: a helping hand for pathway evolution?', *Trends in Biochemical Sciences* **28**(6), 336–341.

Schmien, R., Seiler, K. U., Wassermann, O. & 1974 (n.d.), 'Drug-induced phospholipidosis. I. Lipid composition and chlorphentermine content of rat lung tissue and alveolar macrophages after chronic treatment', **283**, 331–334.

Schomburg, I. (2004), 'BRENDA, the enzyme database: updates and major new developments', *Nucleic Acids Research* **32**(90001), 431D–433.

Schuetz, R., Zamboni, N., Zampieri, M., Heinemann, M. & Sauer, U. (2012), 'Multidimensional optimality of microbial metabolism.', *Science* **336**(6081), 601–604.

Schuster, S., Fell, D. A. & Dandekar, T. (2000), 'A general definition of metabolic pathways useful for systematic organization and analysis of complex metabolic networks.', *Nat. Biotechnol.* **18**(3), 326–332.

Schuster, S., Pfeiffer, T. & Fell, D. A. (2008), 'Is maximization of molar yield in metabolic networks favoured by evolution?', *J. Theor. Biol.* **252**(3), 497–504.

Sheikh, K., Förster, J. & Nielsen, L. K. (2005), 'Modeling Hybridoma Cell Metabolism Using a Generic Genome-Scale Metabolic Model of Mus musculus', *Biotech Prog.* **21**(1), 112–121.

Shen, Y., Liu, J., Estiu, G., Isin, B., Ahn, Y. Y., Lee, D. S., Barabási, A. L., Kapatral, V., Wiest, O. & Oltvai, Z. N. (2010), 'Blueprint for antimicrobial hit discovery targeting metabolic networks', *Proceedings of the National Academy of Sciences* **107**(3), 1082–1087.

Shlomi, T., Cabili, M. N., Herrgård, M. J., Palsson, B. Ø. & Ruppin, E. (2008), 'Network-based prediction of human tissue-specific metabolism.', *Nat. Biotechnol.* **26**(9), 1003–1010.

Shrinet, J., Shastri, J. S., Gaind, R., Bhavesh, N. S. & Sunil, S. (2016), 'Serum metabolomics analysis of patients with chikungunya and dengue mono/co-infections reveals distinct metabolite signatures in the three disease conditions', *Nature Publishing Group* **6**(1), 222.

Smallbone, K., Messiha, H. L., Carroll, K. M., Winder, C. L., Malys, N., Dunn, W. B., Murabito, E., Swainston, N., Dada, J. O., Khan, F., Pir, P., Simeonidis, E., Spasić, I., Wishart, J., Weichart, D., Hayes, N. W., Jameson, D., Broomhead, D. S., Oliver, S. G., Gaskell, S. J., McCarthy, J. E. G., Paton, N. W., Westerhoff, H. V., Kell, D. B. & Mendes, P. (2013), 'A model of yeast glycolysis based on a consistent kinetic characterisation of all its enzymes.', *FEBS Lett.* **587**(17), 2832–2841.

Smallbone, K. & Simeonidis, E. (2009), 'Flux balance analysis: a geometric perspective.', *J. Theor. Biol.* **258**(2), 311–315.

Smallbone, K., Simeonidis, E., Swainston, N. & Mendes, P. (2010), 'Towards a genome-scale kinetic model of cellular metabolism.', *BMC Syst Biol* **4**(1), 6.

Smith, K. F., Goldberg, M., Rosenthal, S., Carlson, L., Chen, J., Chen, C. & Ramachandran, S. (2014), 'Global rise in human infectious disease outbreaks.', *Journal of The Royal Society Interface* **11**(101), 20140950.

Smith, K. F. & Guégan, J.-F. (2010), 'Changing Geographic Distributions of Human Pathogens', *Annual Review of Ecology, Evolution, and Systematics* **41**(1), 231–250.

Soyer, O. S. & O'Malley, M. A. (2013), 'Evolutionary systems biology: what it is and why it matters.', *Bioessays* **35**(8), 696–705.

Stark, R., Pasquel, F., Turcu, A., Pongratz, R. L., Roden, M., Cline, G. W., Shulman, G. I. & Kibbey, R. G. (2009), 'Phosphoenolpyruvate cycling via mitochondrial phosphoenolpyruvate carboxykinase links anaplerosis and mitochondrial GTP with insulin secretion.', *J. Biol. Chem.* **284**(39), 26578–26590.

Steuer, R., Gross, T., Selbig, J. & Blasius, B. (2006), 'Structural kinetic modeling of metabolic networks.', *Proceedings of the National Academy of Sciences* **103**(32), 11868–11873.

Stincone, A., Prigione, A., Cramer, T., Wamelink, M. M. C., Campbell, K., Cheung, E., Olin-Sandoval, V., Grüning, N.-M., Krüger, A., Tauqeer Alam, M., Keller, M. A., Breitenbach, M., Brindle, K. M., Rabinowitz, J. D. & Ralser, M. (2015), 'The return of metabolism: biochemistry and physiology of the pentose phosphate pathway.', *Biol Rev Camb Philos Soc* **90**(3), 927–963.

Stone, R. E. & Tovey, C. A. (1991), 'The Simplex and Projective Scaling Algorithms as Iteratively Reweighted Least Squares Methods', *SIAM Review* **33**(2), 220–237.

Strauss, J. H. & Strauss, E. G. (1994), 'The alphaviruses: gene expression, replication, and evolution.', *Microbiol. Rev.* **58**(4), 806–562.

Stroud, K. A. (1990), Linear Optimisation (Linear Programming), *in* 'Further Engineering Mathematics', Springer New York, New York, NY, pp. 1025–1090.

Swainston, N., Smallbone, K., Hefzi, H., Dobson, P. D., Brewer, J., Hanscho, M., Zielinski, D. C., Ang, K. S., Gardiner, N. J., Gutierrez, J. M., Kyriakopoulos, S., Lakshmanan, M., Li, S., Liu, J. K., Martínez, V. S., Orellana, C. A., Quek, L.-E., Thomas, A., Zanghellini, J., Borth, N., Lee, D.-Y., Nielsen, L. K., Kell, D. B., Lewis, N. E. & Mendes, P. (2016), 'Recon 2.2: from reconstruction to model of human metabolism.', *Metabolomics* **12**(7), 109.

Swann, J., Jamshidi, N., Lewis, N. E. & Winzeler, E. A. (2015), 'Systems analysis of host-parasite interactions', *Wiley Interdiscip Rev Syst Biol Med* **7**(6), 381–400.

Tanaka, M., Yamada, T., Itoh, M., Okuda, S., Goto, S. & Kanehisa, M. (2006), 'Analysis of the differences in metabolic network expansion between prokaryotes and eukaryotes.', *Genome Inform* **17**(1), 230–239.

Tang, B. L. (2012), 'The cell biology of Chikungunya virus infection.', *Cell. Microbiol.* **14**(9), 1354–1363.

Taubenberger, J. K. & Kash, J. C. (2010), 'Influenza Virus Evolution, Host Adaptation, and Pandemic Formation', *Cell Host and Microbe* **7**(6), 440–451.

Thai, M., Graham, N. A., Braas, D., Nehil, M., Komisopoulou, E., Kurdistani, S. K., McCormick, F., Graeber, T. G. & Christofk, H. R. (2014), 'Adenovirus E4ORF1-induced MYC activation promotes host cell anabolic glucose metabolism and virus replication.', *Cell Metabolism* **19**(4), 694–701.

Thai, M., Thaker, S. K., Feng, J., Du, Y., Hu, H., Ting Wu, T., Graeber, T. G., Braas, D. & Christofk, H. R. (2015), 'MYC-induced reprogramming of glutamine catabolism supports optimal virus replication.', *Nature Communications* **6**, 8873.

Thiele, I. & Palsson, B. Ø. (2010), 'A protocol for generating a high-quality genome-scale metabolic reconstruction.', *Nature Protocols* **5**(1), 93–121.

Thiele, I., Swainston, N., Fleming, R. M. T., Hoppe, A., Sahoo, S., Aurich, M. K., Haraldsdottir, H., Mo, M. L., Rolfsson, O., Stobbe, M. D., Thorleifsson, S. G., Agren, R., Bölling, C., Bordel, S., Chavali, A. K., Dobson, P., Dunn, W. B., Endler, L., Hala, D., Hucka, M., Hull, D., Jameson, D., Jamshidi, N., Jonsson, J. J., Juty, N., Keating, S., Nookaew, I., Le Novère, N., Malys, N., Mazein, A., Papin, J. A., Price, N. D., Selkov, E., Sigurdsson, M. I., Simeonidis, E., Sonnenschein, N., Smallbone, K., Sorokin, A., van Beek, J. H. G. M., Weichart, D., Goryanin, I., Nielsen, J., Westerhoff, H. V., Kell, D. B., Mendes, P. & Palsson, B. Ø. (2013), 'A community-driven global reconstruction of human metabolism.', *Nat. Biotechnol.* **31**(5), 419–425.

Timm, A. & Yin, J. (2012), 'Kinetics of virus production from single cells', *Virology* **424**(1), 11–17.

van der Walt, S., Colbert, S. C. & Varoquaux, G. (2011), 'The NumPy Array: A Structure for Efficient Numerical Computation', *Computing in Science & Engineering* **13**(2), 22–30.

Vellai, T. & Vida, G. (1999), 'The origin of eukaryotes: the difference between prokaryotic and eukaryotic cells.', *Proc. Biol. Sci.* **266**(1428), 1571–1577.

Wang, Q.-Y., Bushell, S., Qing, M., Xu, H. Y., Bonavia, A., Nunes, S., Zhou, J., Poh, M. K., Florez de Sessions, P., Niyomrattanakit, P., Dong, H., Hoffmaster, K., Goh, A., Nilar, S., Schul, W., Jones, S., Kramer, L., Compton, T. & Shi,

P.-Y. (2011), 'Inhibition of dengue virus through suppression of host pyrimidine biosynthesis.', *Journal of Virology* **85**(13), 6548–6556.

Wang, Z. & Zhang, J. (2009), 'Abundant indispensable redundancies in cellular metabolic networks.', *Genome Biol Evol* **1**(0), 23–33.

Weiße, A. Y., Oyarzún, D. A., Danos, V. & Swain, P. S. (2015), 'Mechanistic links between cellular trade-offs, gene expression, and growth.', *Proc. Natl. Acad. Sci. U.S.A.* **112**(9), E1038–47.

Wishart, D. S., Knox, C., Guo, A. C., Cheng, D., Shrivastava, S., Tzur, D., Gautam, B. & Hassanali, M. (2008), 'DrugBank: a knowledgebase for drugs, drug actions and drug targets', *Nucleic Acids Research* **36**, D901–D906.

Wolfe, N. D., Dunavan, C. P. & Diamond, J. (2007), 'Origins of major human infectious diseases', *Nature* **447**(7142), 279–283.

Woolhouse, M. E., Adair, K. & Brierley, L. (2014), 'RNA viruses: a case study of the biology of emerging infectious diseases', *... (eds Atlas* .

Woolhouse, M. E. J., Brierley, L., McCaffery, C. & Lycett, S. (2016), 'Assessing the Epidemic Potential of RNA and DNA Viruses', *Emerging Infect. Dis.* **22**(12), 2037–2044.

Yu, C., Hernandez, T., Zheng, H., Yau, S.-C., Huang, H.-H., He, R. L., Yang, J. & Yau, S. S. T. (2013), 'Real time classification of viruses in 12 dimensions.', *PLoS ONE* **8**(5), e64328.

Yu, Y., Clippinger, A. J. & Alwine, J. C. (2011*a*), 'Viral effects on metabolism: changes in glucose and glutamine utilization during human cytomegalovirus infection', *Trends in Microbiology* **19**(7), 360–367.

Yu, Y., Clippinger, A. J. & Alwine, J. C. (2011*b*), 'Viral effects on metabolism: changes in glucose and glutamine utilization during human cytomegalovirus infection', *Trends in Microbiology* **19**(7), 360–367.

Zhu, Y., Yongky, A. & Yin, J. (2009), 'Growth of an RNA virus in single cells reveals a broad fitness distribution.', *Virology* **385**(1), 39–46.

Zomorrodi, A. R. & Segrè, D. (2016), 'Synthetic Ecology of Microbes: Mathematical Models and Applications', *Journal of Molecular Biology* **428**(5), 837–861.

Appendices

A VIRANET PYTHON CODE

Python code for the in-house developed host-virus metabolic modelling package, ViraNet.

A.1 analysis.py

```
#######
# Setup of workspace
import  cobra
from    cobra           import Model, Reaction, Metabolite
from    cobra.flux_analysis.loopless loopless_solution
import  numpy           as np
import  pandas          as pd
import  string
from    viranet         import tools
from    viranet.info    import metDict, aaMets, ntpsMets
#######

#######
# optimise
# Optimises the HVM for host and virus optimal states
# Inputs:
# HVM               Integrated host-virus model
# HostRxn           Host objective reaction, either:
#                   - Index value of reaction in Model.reactions
      [int]
#                   - Reaction ID of the host-objective reaction
      [str]
# solver            [OPTIONAL] Declare solver to use for cobrapy:
    default is cglpk
# fbaType           [OPTIONAL] Declare if FBA is normal 'fba' or
    loopless FBA 'loopless'
#
# Outputs:
# objIdx            Index value for the objective reactions
# hostF             Host-objective optima value
# hostX             Vector of non-objective reaction fluxes for host-
    optimised
# virusF            Virus-objective optima value
# virusX            Vector of non-objective reaction fluxes for virus
    -optimised
```

```
#
def optimise(HVM, HostRxn, solver='cglpk', fbaType='fba'):
    "Optimise HVM"
    # [1] Initial Setup
    # Identify the host objective reaction
    try:
        intTest = int(HostRxn)
        hostIdx = HostRxn
    except:
        for ii in range(len(HVM.reactions)):
            if HostRxn in str(HVM.reactions[ii]):
                hostIdx = ii
    virusIdx    = len(HVM.reactions) - 1    # ViraNet(c) appends virus reaction to end in genHVM.py
    objIdx      = [hostIdx, virusIdx]
    # [2] State Optimisations
    # Identify the reactions
    hostObj     = HVM.reactions[hostIdx]
    virusObj    = HVM.reactions[virusIdx]
    # Record the bounds
    hostLb      = HVM.reactions[hostIdx].lower_bound
    hostUb      = HVM.reactions[hostIdx].upper_bound
    virusLb     = HVM.reactions[virusIdx].lower_bound
    virusUb     = HVM.reactions[virusIdx].upper_bound
    # Host optimisation
    HVM.change_objective(hostObj)
    # Zero-bound virus reaction
    HVM.reactions[virusIdx].lower_bound = 0
    HVM.reactions[virusIdx].upper_bound = 0
    # > Conditional FBA statement
    if 'fbaType' == 'fba':
     # Record Optima
     hostSol     = HVM.optimize(objective_sense='maximize',solver=solver)
     hostX       = hostSol.x                 # Host-optimal flux vector
     hostF       = hostSol.f                 # Optima value for host objective
    elif 'fbaType' == 'loopless'
     # Record Optima
     HVM.solver     = 'cplex'
     HVM.objective  = hostObj.id
     hostSol        = loopless_solution(HVM)
     hostX          = hostSol.x                 # Host-optimal flux vector
     hostFrame      = pd.DataFrame({'reaction': list(HVM.reactions.list_attr('id')),'flux': hostX})
     hostFrame      = hostFrame.sort_values(by='reaction')
     hostF          = hostFrame['flux'][hostFrame['reaction'].isin([hostObj.id])] # Optima value host objective
    # Return virus reaction to correct bounds
    HVM.reactions[virusIdx].lower_bound = virusLb
    HVM.reactions[virusIdx].upper_bound = virusUb
    # Virus optimisation
```

```
77      # Check that virus objective is correct and generated from '
        generation.py'
78      if '_prodrxn_VN' in virusObj.id:
79          # Zero-bound host reaction
80          HVM.reactions[hostIdx].lower_bound = 0
81          HVM.reactions[hostIdx].upper_bound = 0
82          # > Conditional FBA statement
83          if 'fbaType' == 'fba':
84           HVM.change_objective(virusObj)
85           # Record Optima
86           virusSol    = HVM.optimize(objective_sense='maximize',solver
        =solver)
87           virusX      = virusSol.x                # Virus-optimal flux
         vector
88           virusF      = virusSol.f                # Optima value for
        virus objective
89          elif 'fbaType' == 'loopless'
90           # Record Optima
91           HVM.solver     = 'cplex'
92           HVM.objective  = virusObj.id
93           virusSol        = loopless_solution(HVM)
94           virusX          = virusSol.x                    # virus-optimal
         flux vector
95           virusFrame      = pd.DataFrame({'reaction': list(HVM.
        reactions.list_attr('id')),'flux': virusX})
96           virusFrame      = virusFrame.sort_values(by='reaction')
97           virusF          = virusFrame['flux'][virusFrame['reaction'].
        isin([virusObj.id])] # Optima value virus objective
98          # Record virus optima
99          # Return virus reaction to correct bounds
100         HVM.reactions[hostIdx].lower_bound = hostLb
101         HVM.reactions[hostIdx].upper_bound = hostUb
102     else:
103         raise ValueError('Unsupported objective, unable to analyse:
        refer to README')
104     # [3] Outputs
105     return (objIdx, hostF, hostX, virusF, virusX)
106 ########
107
108 ########
109 # differential
110 # Compares the host and virus objective functions for amino acid and
        nucleotide
111 # usage
112 # Inputs:
113 # HVM                 Integrated host-virus model
114 # HostRxn             Host objective reaction, either:
115 #                     - Index value of reaction in Model.reactions
            [int]
116 #                     - Reaction ID of the host-objective reaction
            [str]
117 #
118 # Outputs:
```

```
# rel_difAA,        Relative differential comparison of amino acid and nucleotide
# rel_difNT         usage in the host and virus objective functions
#                   [Relative comparison: amino acids are only considered within
#                   the amino acid fraction; nucleotides are only considered
#                   within the nucleotide fraction]
# abs_difAA,        Absolute differential comparison of amino acid and nucleotide
# abs_difNT         usage in the host and virus objective functions
#                   [Absolute comparison: amino acids and nucleotides are
#                   considered against all macromolecular fractions]
#
def differential(HVM, HostRxn):
    "Differential usage of amino acids and nucleotides"
    # [1] Initial Setup
    # Identify the host objective reaction
    try:
        intTest = int(HostRxn)
        hostIdx = HostRxn
    except:
        for ii in range(len(HVM.reactions)):
            if HostRxn in str(HVM.reactions[ii]):
                hostIdx = ii
    virusIdx    = len(HVM.reactions) - 1    # ViraNet(c) appends virus reaction to end in genHVM.py
    objIdx      = [hostIdx, virusIdx]
    hostID      = HVM.reactions[hostIdx].id
    virusID     = HVM.reactions[virusIdx].id
    # Condition: ensure virus reaction is virus objective
    if '_prodrxn_VN' not in virusID:
        raise ValueError('Unsupported objective, unable to analyse: refer to README')
    # Convert model into an array
    m       = HVM.to_array_based_model()
    # Create data frame
    mFrame  = pd.DataFrame(
        data    = m.S.todense(),
        columns = m.reactions.list_attr("id"),
        index   = m.metabolites.list_attr("id")
    )
    # [2] Differential usage analysis
    # Strip the host and virus objective function stoichiometric coefficients
    hostS   = mFrame[hostID]
    virusS  = mFrame[virusID]
    # Storage variable creation
    hostAA  = np.zeros((20,1))
    hostNT  = np.zeros((4,1))
    virusAA = np.zeros((20,1))
    virusNT = np.zeros((4,1))
    # Loop and collect the stoichiometric coefficients
```

```
    # Amino acids
    for ii in range(len(aaMets)):
        hostAA[ii]  = np.absolute(hostS[metDict[aaMets[ii]]])
        virusAA[ii] = np.absolute(virusS[metDict[aaMets[ii]]])
    # Nucleotides
    for ii in range(len(ntpsMets)):
        if (hostS[metDict['ctp']]) > 0:
            hostNT[ii]  = np.absolute(hostS[metDict[ntpsMets[ii]]])
        else:
            oldMet      = metDict[ntpsMets[ii]]
            tempMet     = oldMet.replace('tp','mp')
            hostNT[ii]  = np.absolute(hostS[tempMet])
        if ntpsMets[ii] is not "atp":
            virusNT[ii] = np.absolute(virusS[metDict[ntpsMets[ii]]])
        else:
            virusNT[ii] = np.absolute(virusS[metDict[ntpsMets[ii]]])
    - np.absolute(virusS[metDict['adp']])
        # [3] Create output
        # Dictionary to hold metabolites with corresponding values
        rel_dif = dict()
        abs_dif = dict()
        # Relative difference
        for ii in range(len(aaMets)):
            rdifH  = hostAA[ii] / np.sum(hostAA)
            rdifV  = virusAA[ii] / np.sum(virusAA)
            rdifX  = (rdifV / rdifH) - 1
            rdifX  = rdifX.tolist()
            rdifX  = rdifX[0]
            rel_dif[aaMets[ii]] = rdifX
        for ii in range(len(ntpsMets)):
            rdifH  = hostNT[ii] / np.sum(hostNT)
            rdifV  = virusNT[ii] / np.sum(virusNT)
            rdifX  = (rdifV / rdifH) - 1
            rdifX  = rdifX.tolist()
            rdifX  = rdifX[0]
            rel_dif[ntpsMets[ii]] = rdifX
        # Absolute difference
        for ii in range(len(aaMets)):
            adifH  = hostAA[ii] / np.sum(hostS)
            adifV  = virusAA[ii] / np.sum(virusS)
            adifX  = (adifV / adifH) - 1
            adifX  = adifX.tolist()
            adifX  = adifX[0]
            abs_dif[aaMets[ii]] = adifX
        for ii in range(len(ntpsMets)):
            adifH  = hostNT[ii] / np.sum(hostS)
            adifV  = virusNT[ii] / np.sum(virusS)
            adifX  = (adifV / adifH) - 1
            adifX  = adifX.tolist()
            adifX  = adifX[0]
            abs_dif[ntpsMets[ii]] = adifX
    # Return values
    return(rel_dif, abs_dif)
#######
```

```
218
219 #######
220 # compare
221 # Compares the flux distribution of a host-optimal state with that of
        a virus-
222 # optimal state
223 # Inputs:
224 # objIdx              Index value for the objective reactions
225 # hostX               Vector of non-objective reaction fluxes for host-
        optimised
226 # virusX              Vector of non-objective reaction fluxes for virus
        -optimised
227 #
228 # Outputs:
229 # hvmComp             Comparison of HOS and VOS flux values (as % total
        flux)
230 # hvmStat             Reaction stats for HOS and VOS states:
231 #                     - Upregulated
232 #                     - Downregulated
233 #                     - Activated
234 #                     - Inactivated
235 #                     - Reversed
236
237 def compare(objIdx,hostX,virusX):
238     "Host-Virus Comparison"
239     # [1] Initial Setup
240     # Numpy Conversion
241     hostX   = np.array(hostX)
242     virusX  = np.array(virusX)
243     # Remove the objective reactions from both flux vectors
244     hostXd  = np.delete(hostX,objIdx)
245     virusXd = np.delete(virusX,objIdx)
246     # Convert flux vectors to absolute and normalise to summation of
        vector
247     pHOS    = (hostXd / sum(np.absolute(hostX))) * 100
248     pVOS    = (virusXd / sum(np.absolute(virusX))) * 100
249     # Convert to suitable numpy array
250     pHOS    = np.array(pHOS)
251     pVOS    = np.array(pVOS)
252     # [2] Reaction Statistics
253     # Variables for calculations
254     tol     = 1.05                                                  #
        Regulated tolerance
255     e       = 1e-06                                                 #
        Threshold for 'on'
256     ne      = e * -1                                                #
        Negative threshold
257     inf     = np.inf
258     # Reaction states: Upregulated; Downregulated; Activated;
        Inactivated; Reversed
259     urRxns = 0
260     drRxns = 0
261     avRxns = 0
262     iaRxns = 0
```

```
263     reRxns = 0
264     # Initiate loop
265     for ii in range(len(pHOS)):
266         # Upregulated
267         if (pVOS[ii] > e) and (pHOS[ii] > e) and ((pVOS[ii] / pHOS[ii
    ]) > tol) and ((pVOS[ii] / pHOS[ii]) < inf):
268             urRxns += 1
269         elif (pVOS[ii] < ne) and (pHOS[ii] < ne) and ((pVOS[ii] /
    pHOS[ii]) > tol) and ((pVOS[ii] / pHOS[ii]) < inf):
270             urRxns += 1
271         # Downregulated
272         if (pVOS[ii] > e) and (pHOS[ii] > e) and ((pHOS[ii] / pVOS[ii
    ]) > tol) and ((pHOS[ii] / pVOS[ii]) < inf):
273             drRxns += 1
274         elif (pVOS[ii] < ne) and (pHOS[ii] < ne) and ((pHOS[ii] /
    pVOS[ii]) > tol) and ((pHOS[ii] / pVOS[ii]) < inf):
275             drRxns += 1
276         # Activated
277         if (np.absolute(pVOS[ii]) > e) and (np.absolute(pHOS[ii]) < e
    ):
278             avRxns += 1
279         # Inactivated
280         if (np.absolute(pVOS[ii]) < e) and (np.absolute(pHOS[ii]) > e
    ):
281             iaRxns += 1
282         # Reversed
283         if (pVOS[ii] > e and pHOS[ii] < ne) or (pVOS[ii] < ne and
    pHOS[ii] > e):
284             reRxns +=1
285     # [3] Outputs
286     # hvmComp
287     hvmComp      = np.vstack((pHOS,pVOS))
288     hvmComp      = hvmComp.transpose()
289     # hvmStat
290     hvmStat      = [urRxns,drRxns,avRxns,iaRxns,reRxns]
291 
292     return(hvmComp,hvmStat,hostXd,virusXd)
293 ########
294 
295 ########
296 # varability
297 # Optimises the HVM for host and virus optimal states, using flux
    variability
298 # analysis
299 # Inputs:
300 # HVM               Integrated host-virus model
301 # HostRxn           Host objective reaction, either:
302 #                   - Index value of reaction in Model.reactions
        [int]
303 #                   - Reaction ID of the host-objective reaction
        [str]
304 # Optional Inputs
305 # solver            Declare solver to use for cobrapy: default is
    cglpk
```

```
306 #
307 # Outputs:
308 # hostFVA            Vector of non-objective reaction fluxes for host-
    optimised  [maximum | minimum]
309 # virusFVA           Vector of non-objective reaction fluxes for virus
    -optimised [maximum | minimum]
310 #
311 def variability(HVM, HostRxn, solver='cglpk'):
312     "Variability of HVM"
313     # [1] Initial Setup
314     # Identify the host objective reaction
315     try:
316         intTest = int(HostRxn)
317         hostIdx = HostRxn
318     except:
319         for ii in range(len(HVM.reactions)):
320             if HostRxn in str(HVM.reactions[ii]):
321                 hostIdx = ii
322     virusIdx    = len(HVM.reactions) - 1    # ViraNet(c) appends
    virus reaction to end in genHVM.py
323     objIdx      = [hostIdx, virusIdx]
324     # [2] State Optimisations
325     ######
326     # Host optimisation #
327     hostObj     = HVM.reactions[hostIdx]
328     HVM.change_objective(hostObj)
329     # Ensure no flux can go through the virus reaction
330     # Store the bounds
331     virusLb     = HVM.reactions[virusIdx].lower_bound
332     virusUb     = HVM.reactions[virusIdx].upper_bound
333     HVM.reactions[virusIdx].lower_bound = 0
334     HVM.reactions[virusIdx].upper_bound = 0
335     # FVA
336     varHost     = cobra.flux_analysis.flux_variability_analysis(HVM,
    solver=solver)
337     # Return virus objective bounds
338     HVM.reactions[virusIdx].lower_bound = virusLb
339     HVM.reactions[virusIdx].upper_bound = virusUb
340     ######
341     # Virus optimisation #
342     virusObj    = HVM.reactions[virusIdx]
343     # Check that virus objective is correct and generated from '
    generation.py'
344     if '_prodrxn_VN' in virusObj.id:
345         HVM.change_objective(virusObj)
346         # Ensure no flux can go through the host reaction
347         # Store the bounds
348         hostLb      = HVM.reactions[hostIdx].lower_bound
349         hostUb      = HVM.reactions[hostIdx].upper_bound
350         HVM.reactions[hostIdx].lower_bound = 0
351         HVM.reactions[hostIdx].upper_bound = 0
352         # FVA
353         varVirus    = cobra.flux_analysis.flux_variability_analysis(
    HVM, solver=solver)
```

```
354         # Return host objective bounds
355         HVM.reactions[hostIdx].lower_bound = hostLb
356         HVM.reactions[hostIdx].upper_bound = hostUb
357     else:
358         raise ValueError('Unsupported objective, unable to analyse:
    refer to README')
359     ######
360     # [3] Outputs
361     # Create of data frames
362     hostFVA   = pd.DataFrame.from_dict(varHost)
363     hostFVA   = hostFVA.transpose()
364     virusFVA  = pd.DataFrame.from_dict(varVirus)
365     virusFVA  = virusFVA.transpose()
366     # Return
367     return (hostFVA, virusFVA)
368
369 #######
370
371 #######
372 # knockout
373 # Analysis of the effect of a single-reaction knockout on virus
    production
374 # Inputs:
375 # HVM               Integrated host-virus model
376 # solver            [OPTIONAL] Declare solver to use for cobrapy:
    default is cglpk
377 #
378 # Outputs:
379 # koVirus           Vector of virus optima values for single-reaction
    knockout
380
381 def knockout(HVM, HostRxn, solver='cglpk'):
382     "Reaction Knockouts"
383     # [1] Initial Setup
384     # Identify the host objective reaction
385     try:
386         intTest = int(HostRxn)
387         hostIdx = HostRxn
388     except:
389         for ii in range(len(HVM.reactions)):
390             if HostRxn in str(HVM.reactions[ii]):
391                 hostIdx = ii
392     # Ensure virus is objective
393     HVM.change_objective(HVM.reactions[-1])
394     # Variable creation to hold virus optima
395     koVirus     = np.zeros((len(HVM.reactions),1))
396     # [2] Knockout Analysis
397     # Record the host bounds
398     hostLb      = HVM.reactions[hostIdx].lower_bound
399     hostUb      = HVM.reactions[hostIdx].upper_bound
400     # Initiate loop
401     for ii in range(len(HVM.reactions)):
402         # Store the bounds of the reaction [ii]
403         rxnLb = HVM.reactions[ii].lower_bound
```

```
404         rxnUb = HVM.reactions[ii].upper_bound
405         # Alter bounds to zero
406         HVM.reactions[ii].lower_bound = 0
407         HVM.reactions[ii].upper_bound = 0
408         # Ensure the host objective is set to zero bounds
409         HVM.reactions[hostIdx].lower_bound = 0
410         HVM.reactions[hostIdx].upper_bound = 0
411         # Record and store the virus optima
412         vSol = HVM.optimize(objective_sense='maximize',solver=solver)
413         koVirus[ii] = vSol.f
414         # Return reaction [ii] to it's original bounds
415         HVM.reactions[ii].lower_bound = rxnLb
416         HVM.reactions[ii].upper_bound = rxnUb
417     # Return host bounds to original
418     HVM.reactions[hostIdx].lower_bound = hostLb
419     HVM.reactions[hostIdx].upper_bound = hostUb
420     # [3] Outputs
421     return koVirus
422 #######
423
424 #######
425 # enforcement
426 # Analysis of the effect of a single-reaction host-derived constraint
      enforced
427 # for a virus-optimised system
428 # Inputs:
429 # HVM               Integrated host-virus model
430 # hostX             Vector of non-objective reaction fluxes for host-
    optimised
431 # Optional Inputs
432 # solver            Declare solver to use for cobrapy: default is
    cglpk
433 # usefva            Use the Host FBA result (False) or use the FVA
    range as
434 #                   the additional constraint (True)
435 # userange          Use the Host FVA results for enforcement analysis
    , either the
436 #                   median FVA value for both bounds (False) or the
    min / max FVA
437 #                   as the reaction bounds (True)
438 # HostRxn           Host objective reaction, either:
439 #                   - Index value of reaction in Model.reactions
        [int]
440 #                   - Reaction ID of the host-objective reaction
        [str]
441 # Outputs:
442 # enfVirus          Vector of virus optima values with additional
    host-constraint
443 # Optional Outputs:
444 # maxEnfBound       Maximum enf bound
445 # minEnfBound       Minimum enf bound
446
447 def enforce(HVM,hostX,HostRxn,solver='cglpk',usefva=False,userange=
    False):
```

```
448     "Host-derived Enforcement"
449     # [1] Initial Setup
450     # Identify virus reaction
451     virusIdx    = len(HVM.reactions) - 1    # ViraNet(c) appends
    virus reaction to end in genHVM.py
452     virusObj    = HVM.reactions[virusIdx]
453     # FVA Conditional
454     if usefva == False:
455         # Ensure virus is objective
456         HVM.change_objective(HVM.reactions[-1])
457         # Numpy Conversion
458         hostX   = np.array(hostX)
459     elif usefva == True:
460         # Ensure host is objective
461         # Identify the host objective reaction
462         try:
463             intTest = int(HostRxn)
464             hostIdx = HostRxn
465         except:
466             for ii in range(len(HVM.reactions)):
467                 if HostRxn in str(HVM.reactions[ii]):
468                     hostIdx = ii
469         hostObj     = HVM.reactions[hostIdx]
470         HVM.change_objective(hostObj)
471         # Record the bounds
472         hostLb      = HVM.reactions[hostIdx].lower_bound
473         hostUb      = HVM.reactions[hostIdx].upper_bound
474         virusLb     = HVM.reactions[virusIdx].lower_bound
475         virusUb     = HVM.reactions[virusIdx].upper_bound
476         # Host optimisation
477         HVM.change_objective(hostObj)
478         # Zero-bound virus reaction
479         HVM.reactions[virusIdx].lower_bound = 0
480         HVM.reactions[virusIdx].upper_bound = 0
481         # Perform flux varability analysis
482         varHost     = cobra.flux_analysis.flux_variability_analysis(
    HVM, solver=solver)
483         # Return virus reaction to correct bounds
484         HVM.reactions[virusIdx].lower_bound = virusLb
485         HVM.reactions[virusIdx].upper_bound = virusUb
486     # Variable creation to hold virus optima
487     HVM.change_objective(virusObj)
488     enfVirus    = np.zeros((len(hostX),1))
489     # [2] Enforcement Analysis
490     # Initiate loop
491     ########################################
492     # USEFVA #
493     # False conditional
494     if usefva == False:
495         # Record the host bounds
496         hostLb      = HVM.reactions[hostIdx].lower_bound
497         hostUb      = HVM.reactions[hostIdx].upper_bound
498         # Initiate loop
499         for ii in range(len(HVM.reactions)):
```

```
500             # Store the bounds of the reaction [ii]
501             rxnLb = HVM.reactions[ii].lower_bound
502             rxnUb = HVM.reactions[ii].upper_bound
503             # Ensure the host objective is set to zero bounds
504             HVM.reactions[hostIdx].lower_bound = 0
505             HVM.reactions[hostIdx].upper_bound = 0
506             # Alter reacton [ii] bounds to match host-derived flux:
    hostX[ii]
507             HVM.reactions[ii].lower_bound = hostX[ii]
508             HVM.reactions[ii].upper_bound = hostX[ii]
509             # Record and store the virus optima
510             vSol = HVM.optimize(objective_sense='maximize',solver=
    solver)
511             enfVirus[ii] = vSol.f
512             # Return reaction [ii] to it's original bounds
513             HVM.reactions[ii].lower_bound = rxnLb
514             HVM.reactions[ii].upper_bound = rxnUb
515         # Return host bounds to original
516         HVM.reactions[hostIdx].lower_bound = hostLb
517         HVM.reactions[hostIdx].upper_bound = hostUb
518     # True conditional
519     elif usefva == True:
520         # Record the host bounds
521         hostLb       = HVM.reactions[hostIdx].lower_bound
522         hostUb       = HVM.reactions[hostIdx].upper_bound
523         ########################################
524         # USERANGE #
525         # Conditional: Median of FVA range
526         if userange == False:
527             for ii in range(len(HVM.reactions)):
528                 # Store the bounds of the reaction [ii]
529                 rxnLb = HVM.reactions[ii].lower_bound
530                 rxnUb = HVM.reactions[ii].upper_bound
531                 # Alter reacton [ii] bounds to match median host-
    derived flux from FVA: varHost[ii]
532                 maxFVA       = varHost[HVM.reactions[ii].id]['maximum'
    ]
533                 minFVA       = varHost[HVM.reactions[ii].id]['minimum'
    ]
534                 medianFVA    = maxFVA - ((maxFVA - minFVA) / 2)
535                 HVM.reactions[ii].lower_bound = medianFVA
536                 HVM.reactions[ii].upper_bound = medianFVA
537                 # Ensure the host objective is set to zero bounds
538                 HVM.reactions[hostIdx].lower_bound = 0
539                 HVM.reactions[hostIdx].upper_bound = 0
540                 # Record and store the virus optima
541                 vSol = HVM.optimize(objective_sense='maximize',solver
    =solver)
542                 enfVirus[ii] = vSol.f
543                 # Return reaction [ii] to it's original bounds
544                 HVM.reactions[ii].lower_bound = rxnLb
545                 HVM.reactions[ii].upper_bound = rxnUb
546         # Conditional: Max / min bounds from FVA for reaction
547         elif userange == True:
```

```
548             # Pass flux variability analysis results to tools.
    rangeCalculator
549             (enfVirus, maxEnfBound, minEnfBound) = tools.
    rangeCalculator(HVM, hostIdx, virusIdx, solver)
550         # Return host bounds to original
551         HVM.reactions[hostIdx].lower_bound = hostLb
552         HVM.reactions[hostIdx].upper_bound = hostUb
553         ##########################################
554     ##########################################
555     # [3] Outputs
556     return (enfVirus,maxEnfBound,minEnfBound)
557 ########
```

Listing 1: Viranet Sub-Package: analysis.py

A.2 generation.py

```
#######
# generation.py generates the virus biomass objective function (VBOF), and the
# host-virus integrated model (HVM) for a given virus genome file (virusGB) and
# a given host metabolic model (hostModel) file (.mat,.xml)
#######
# Setup of workspace
import cobra
from cobra import Model, Reaction, Metabolite
import numpy as np
from viranet.info import metDict, ntpsDict, aaDict, miscDict, ntpsMets, aaMets, N_A, k_atp, k_ppi
import re

# Definitions
# Virus locations
# This indicate the 6 starting amino acids of the non-structural polyprotein
# for the supported flavivurses. These are used to determine the seperation
# of structural and non-structural polyproteins
DENVloc = 'DSGCVV'
ZIKVloc = 'DVGCSV'
# Metabolite definitions for final objective reaction creation
# This creates a mapping between the python-cobrapy and the ViraNet function
atp_c = Metabolite(metDict['atp'])
ctp_c = Metabolite(metDict['ctp'])
gtp_c = Metabolite(metDict['gtp'])
utp_c = Metabolite(metDict['utp'])
ala_c = Metabolite(metDict['A'])
arg_c = Metabolite(metDict['R'])
asn_c = Metabolite(metDict['N'])
asp_c = Metabolite(metDict['D'])
cys_c = Metabolite(metDict['C'])
gln_c = Metabolite(metDict['Q'])
glu_c = Metabolite(metDict['E'])
gly_c = Metabolite(metDict['G'])
his_c = Metabolite(metDict['H'])
ile_c = Metabolite(metDict['I'])
leu_c = Metabolite(metDict['L'])
lys_c = Metabolite(metDict['K'])
met_c = Metabolite(metDict['M'])
phe_c = Metabolite(metDict['F'])
pro_c = Metabolite(metDict['P'])
ser_c = Metabolite(metDict['S'])
thr_c = Metabolite(metDict['T'])
trp_c = Metabolite(metDict['W'])
tyr_c = Metabolite(metDict['Y'])
val_c = Metabolite(metDict['V'])
adp_c = Metabolite(metDict['adp'])
```

```
47 h2o_c = Metabolite(metDict['h2o'])
48 h_c   = Metabolite(metDict['h'])
49 pi_c  = Metabolite(metDict['Pi'])
50 ppi_c = Metabolite(metDict['PPi'])
51 #######
52 # Generation of the virus biomass objective function
53 # Inputs:
54 # VirusGB   User-supplied GenBank file (NCBI) for desired virus [.gb,
       .txt]
55
56 # Outputs:
57 # VBOF      Virus biomass objective function for desired virus
58
59 def VBOF(VirusGB):
60     "Generate_VBOF"
61
62     # [1] Initial Setup
63     # Open virus file and parse contents
64     with open(VirusGB,'rU') as vf:
65         virusFile     = vf.readlines()
66     for ii in range(len(virusFile)):
67         virusFile[ii] = virusFile[ii].rstrip('\n')
68     # Identify virus genera and define virusMethod
69     # virusMethod denotes the method of VBOF generation to use, based
       upon the
70     # virus genera
71     virusData = str(virusFile)
72     if "alphavirus".lower()     in virusData.lower():
73         virusMethod = 1
74     elif "flavivirus".lower()   in virusData.lower():
75         virusMethod = 2
76     else:
77         raise ValueError('Unsupported virus, unable to create VBOF:
      refer to README')
78     # VBOF Parameters: virusMethod dependent
79     # Alphavirus
80     if virusMethod == 1:
81         # Copy number for viral genome                        [Cg]
                # Source: Strauss, J. H., & Strauss, E. G. (1994).
82         Cg  = 1
83         # Copy number for viral structural polyprotein        [Csp]
                # Source: Strauss, J. H., & Strauss, E. G. (1994).
84         Csp = 240
85         # Copy number for viral nonstructural polyprotein     [Cnp]
                # Source: Strauss, J. H., & Strauss, E. G. (1994).
86         Cnp = 1
87     # Flavivirus
88     elif virusMethod == 2:
89         # Copy number for viral genome                        [Cg]
                # Source: Mukhopadhyay, S., Kuhn, R. J., & Rossmann, M.
       G. (2005).
90         Cg  = 1
91         # Copy number for viral structural polyprotein        [Csp]
                # Source: Mukhopadhyay, S., Kuhn, R. J., & Rossmann, M.
```

```
    G. (2005).
92          Csp = 180
93          # Copy number for viral nonstructural polyprotein    [Cnp]
                  # Source: Mukhopadhyay, S., Kuhn, R. J., & Rossmann, M.
    G. (2005).
94          Cnp = 1
95      # Virus Name
96      # FUTURE UPDATE: Definition via blatimore classification and
    genera / species
97      if "chikungunya".lower() in virusData.lower():
98          virusName = "CHIKV"
99          virusFull = "Chikungunya Virus"
100     elif "semliki".lower() in virusData.lower():
101         virusName = "SFV"
102         virusFull = "Semliki Forest Virus"
103     elif "sindbis".lower() in virusData.lower():
104         virusName = "SINV"
105         virusFull = "Sindbis Virus"
106     elif "dengue".lower() in virusData.lower():
107         virusName = "DENV"
108         virusFull = "Dengue Virus"
109         flavMeth  = 1
                  # Indicator for nonstructural start
110     elif "zika".lower() in virusData.lower():
111         virusName = "ZIKV"
112         virusFull = "Zika Virus"
113         flavMeth  = 2
                  # Which Virus Location paramter to use
114     elif "eastern".lower() in virusData.lower():
115         virusName = "EEEV"
116         virusFull = "Eastern Equine Encephalitis Virus"
117     elif "western".lower() in virusData.lower():
118         virusName = "WEEV"
119         virusFull = "Western Equine Encephalitis Virus"
120     elif "venezuelan".lower() in virusData.lower():
121         virusName = "VEEV"
122         virusFull = "Venezuelan Equine Encephalitis Virus"
123     else:
124         raise ValueError('Unsupported virus, unable to create VBOF:
    refer to README')
125
126     # [2] Sequence Identification
127     # Genome Sequence: initial step is to identify start/end
    positions in the virus file
128     startG  = [jj for jj, s in enumerate(virusFile) if 'ORIGIN' in s]
129     endG    = [jj for jj, s in enumerate(virusFile) if '//' in s]
130     startG  = int(''.join(map(str,startG)))
131     endG    = int(''.join(map(str,endG)))
132     startG  = startG + 1
133     # Store genome sequence
134     regex           = re.compile('[^a-zA-Z]')
135     virusGenome     = str(''.join(virusFile[startG:endG]))
136     virusGenome     = regex.sub('',virusGenome)
137     # Polyprotein sequences: split into structural and nonstructural
```

```
138     # This step is virus genera dependent (supported viruses only,
        see README)
139     # Alphavirus
140     if virusMethod == 1:
141         # Structural polyprotein identification
142         tempS1          = [jj for jj, s in enumerate(virusFile) if '/
        product="structural polyprotein"' in s]
143         tempStruct      = virusFile[tempS1[0]:]
144         tempS2          = [jj for jj, s in enumerate(tempStruct) if '
        /translation' in s]
145         tempStruct      = tempStruct[tempS2[0]:]
146         tempS3          = [jj for jj, s in enumerate(tempStruct) if '
        gene' in s]
147         tempStruct      = tempStruct[:tempS3[0]]
148         # Clean-up and Store
149         structReg       = re.compile('/translation=')
150         virusStruct     = str(tempStruct)
151         virusStruct     = structReg.sub('',virusStruct)
152         virusStruct     = regex.sub('',virusStruct)
153         # Nonstructural polyprotein identification
154         tempN1          = [jj for jj, s in enumerate(virusFile) if '/
        product="nonstructural polyprotein"' in s]
155         tempNP          = virusFile[tempN1[0]:]
156         tempN2          = [jj for jj, s in enumerate(tempNP) if '/
        translation' in s]
157         tempNP          = tempNP[tempN2[0]:]
158         tempN3          = [jj for jj, s in enumerate(tempNP) if 'gene
        ' in s]
159         tempNP          = tempNP[:tempN3[0]]
160         # Clean-up and Store
161         npReg           = re.compile('/translation=')
162         virusNonStruct  = str(tempNP)
163         virusNonStruct  = npReg.sub('',virusNonStruct)
164         virusNonStruct  = regex.sub('',virusNonStruct)
165     # Flavivirus
166     elif virusMethod == 2:
167         # Polyprotein identification
168         tempP1          = [jj for jj, s in enumerate(virusFile) if '/
        product="flavivirus polyprotein"' in s]
169         tempPol         = virusFile[tempP1[0]:]
170         tempP2          = [jj for jj, s in enumerate(tempPol) if '/
        translation' in s]
171         tempPol         = tempPol[tempP2[0]:]
172         tempP3          = [jj for jj, s in enumerate(tempPol) if '
        gene' in s]
173         tempPol         = tempPol[:tempP3[0]-1]
174         npReg           = re.compile('/translation=')
175         virusPoly       = str(tempPol)
176         virusPoly       = npReg.sub('',virusPoly)
177         virusPoly       = regex.sub('',virusPoly)
178         # Structural and nonstructural polyprotein identification
179         # flavMeth conditional variable: [1] Dengue virus; [2] Zika
        virus
180         if flavMeth == 1:
```

```
181             nsInd           = re.search(DENVloc, virusPoly).start()
182             virusNonStruct  = virusPoly[nsInd:]
183             virusStruct     = virusPoly[:nsInd]
184         elif flavMeth == 2:
185             nsInd           = re.search(ZIKVloc, virusPoly).start()
186             virusNonStruct  = virusPoly[nsInd:]
187             virusStruct     = virusPoly[:nsInd]
188     # No supported virus detected
189     else:
190         raise ValueError('Unsupported virus, unable to create VBOF:
    refer to README')
191
192     # [3] Precursor frequency
193     # Genome                                [Nucleotides]
194     countA  = virusGenome.count('a')
195     countC  = virusGenome.count('c')
196     countG  = virusGenome.count('g')
197     countU  = virusGenome.count('t')    # Base 'T' is psuedo for base
     'U'
198     antiA   = countU
199     antiC   = countG
200     antiG   = countC
201     antiU   = countA
202     # Structural polyprotein            [Amino Acids]
203     structCount     = np.zeros((20,1))
204     for ii in range(len(aaMets)):
205         structCount[ii,0] = virusStruct.count(aaMets[ii])
206     # Nonstructural polyprotein         [Amino Acids]
207     nonstructCount  = np.zeros((20,1))
208     for ii in range(len(aaMets)):
209         nonstructCount[ii,0] = virusNonStruct.count(aaMets[ii])
210     # Count summation
211     totNTPS     = (Cg * (countA + countC + countG + countU + antiA +
    antiC + antiG + antiU))
212     totAA       = (structCount * Csp) + (nonstructCount * Cnp)
213
214     # [4] VBOF Calculations
215     # Nucleotides
216     # mol.ntps/mol.virus
217     V_a = (Cg*(countA + antiA))
218     V_c = (Cg*(countC + antiC))
219     V_g = (Cg*(countG + antiG))
220     V_u = (Cg*(countU + antiU))
221     # g.ntps/mol.virus
222     G_a = V_a * ntpsDict["atp"]
223     G_c = V_c * ntpsDict["ctp"]
224     G_g = V_g * ntpsDict["gtp"]
225     G_u = V_u * ntpsDict["utp"]
226     # Amino Acids
227     # mol.aa/mol.virus
228     V_aa    = np.zeros((20,1))
229     for ii in range(len(aaMets)):
230         V_aa[ii,0] = totAA[ii]
231     # g.a/mol.virus
```

```
232     G_aa    = np.zeros((20,1))
233     for ii in range(len(aaMets)):
234         G_aa[ii,0] = V_aa[ii] * aaDict[aaMets[ii]]
235     # Total genomic and proteomic molar mass
236     M_v     = (G_a + G_c + G_g + G_u) + G_aa.sum()
237     # Stoichiometric coefficients
238     # Nucleotides [mmol.ntps/g.virus]
239     S_atp = 1000 * (V_a/M_v)
240     S_ctp = 1000 * (V_c/M_v)
241     S_gtp = 1000 * (V_g/M_v)
242     S_utp = 1000 * (V_u/M_v)
243     # Amino acids [mmol.aa/g.virus]
244     S_aa    = np.zeros((20,1))
245     for ii in range(len(aaMets)):
246         S_aa[ii] = 1000 * (V_aa[ii]/M_v)
247     # Energy requirements
248     # Genome: Phosphodiester bond formation products [Pyrophosphate]
249     genTemp = (((countA + countC + countG + countU) * k_ppi) - k_ppi)
250     genRep  = (((antiA + antiC + antiG + antiU) * k_ppi) - k_ppi)
251     genTot  = genTemp + genRep
252     V_ppi   = genTot
253     S_ppi   = 1000 * (V_ppi/M_v)
254     # Protome: Peptide bond formation [ATP + H2O]
255     # Note: ATP used in this process is denoated as ATPe/Ae [e =
        energy version]
256     spAe    = ((structCount.sum() * k_atp) - k_atp)
257     npAe    = ((nonstructCount.sum() * k_atp) - k_atp)
258     ppTot   = (Csp * spAe) + (Cnp * npAe)
259     V_Ae    = ppTot
260     S_Ae    = 1000 * (V_Ae/M_v)
261
262     # [5] VBOF Reaction formatting and output
263     # Left-hand terms: Nucleotides
264     # Note: ATP term is a summation of genome and energy requirements
265     S_ATP   = (S_atp + S_Ae) * -1
266     S_CTP   = S_ctp * -1
267     S_GTP   = S_gtp * -1
268     S_UTP   = S_utp * -1
269     # Left-hand terms: Amino Acids
270     S_AA    = S_aa * -1
271     S_AAf   = dict()
272     for ii in range(len(aaMets)):
273         S_AAf[aaMets[ii]] = S_AA[ii,0]
274     # Left-hand terms: Energy Requirements
275     S_H2O   = S_Ae * -1
276     # Right-hand terms: Energy Requirements
277     S_ADP   = S_Ae
278     S_Pi    = S_Ae
279     S_H     = S_Ae
280     S_PPi   = S_ppi
281     # Create reaction output
282     reaction_name       = virusName + '_prodrxn_VN'
283     virus_reaction      = Reaction(reaction_name)
284     virus_reaction.name = virusFull + ' production reaction (created:
```

```
    ViraNet(c))'
    virus_reaction.subsystem                = 'Virus Production'
    virus_reaction.lower_bound              = 0
    virus_reaction.upper_bound              = 1000
    virus_reaction.objective_coefficient    = 0
    virus_reaction.add_metabolites(({
        atp_c: S_ATP,
        ctp_c: S_CTP,
        gtp_c: S_GTP,
        utp_c: S_UTP,
        ala_c: S_AAf['A'],
        arg_c: S_AAf['R'],
        asn_c: S_AAf['N'],
        asp_c: S_AAf['D'],
        cys_c: S_AAf['C'],
        gln_c: S_AAf['Q'],
        glu_c: S_AAf['E'],
        gly_c: S_AAf['G'],
        his_c: S_AAf['H'],
        ile_c: S_AAf['I'],
        leu_c: S_AAf['L'],
        lys_c: S_AAf['K'],
        met_c: S_AAf['M'],
        phe_c: S_AAf['F'],
        pro_c: S_AAf['P'],
        ser_c: S_AAf['S'],
        thr_c: S_AAf['T'],
        trp_c: S_AAf['W'],
        tyr_c: S_AAf['Y'],
        val_c: S_AAf['V'],
        h2o_c: S_H2O,
        adp_c: S_ADP,
        pi_c:  S_Pi,
        h_c:   S_H,
        ppi_c: S_PPi}))

    # DEBUG COMMENT
    # print("")
    # print("VBOF Reaction Information")
    # print(virus_reaction)
    # for ii in virus_reaction.metabolites:
    #     print("%9s : %s" % (ii.id, ii.values))
    # print(virus_reaction)

    # [6] Output return variables
    return (virus_reaction)

#######

#######
# Generation of the human-virus intergrated model (HVM): generated
    from a
# user-supplied host stoichiometric model and a given VBOF (generated
    : VBOF.py)
```

```
# Inputs:
# Model          User-supplied model file for desired host   [.mat,.xml]
# VBOF           Virus biomass objective function

# Outputs:
# hvm            Integrated host-virus model

def HVM(Model,VBOF):
    "Generate_HVM"

    # [1] Intial Setup
    # File-type dependent loading
    if ".mat".lower() in Model.lower():
        hostModel = cobra.io.load_matlab_model(Model)
    elif ".xml".lower() in Model.lower():
        hostModel  = cobra.io.read_sbml_model(Model)
    else:
        raise ValueError('Unsupported file type, unable to load model: see README')
    # [2] VBOF integration
    hostModel.add_reaction(VBOF)
    # [3] Alter the bounds of VBOF
    hostModel.reactions[-1].reversibility = False
    return (hostModel)
#######
```

Listing 2: Viranet Sub-Package: generation.py

A.3 info.py

```
# VIRANET DEFINITIONS #
# Metabolite (verbose to model) definitions
metDict = {
    'atp': 'atp[c]',        # ATP,              ChEBI 15422
    'ctp': 'ctp[c]',        # CTP,              ChEBI 17677
    'gtp': 'gtp[c]',        # GTP,              ChEBI 15996
    'utp': 'utp[c]',        # UTP,              ChEBI 15713
    'A': 'ala-L[c]',        # Alaline,          ChEBI 16977
    'R': 'arg-L[c]',        # Arginine,         ChEBI 16467
    'N': 'asn-L[c]',        # Asparagine,       ChEBI 17196
    'D': 'asp-L[c]',        # Aspartate,        ChEBI 17053
    'C': 'cys-L[c]',        # Cysteine,         ChEBI 17561
    'Q': 'gln-L[c]',        # Glutamine,        ChEBI 18050
    'E': 'glu-L[c]',        # Glutamate,        ChEBI 16015
    'G': 'gly[c]',          # Glycine,          ChEBI 15428
    'H': 'his-L[c]',        # Histidine,        ChEBI 15971
    'I': 'ile-L[c]',        # Isoleucine,       ChEBI 17191
    'L': 'leu-L[c]',        # Leucine,          ChEBI 15603
    'K': 'lys-L[c]',        # Lysine,           ChEBI 18019
    'M': 'met-L[c]',        # Methionine,       ChEBI 16643
    'F': 'phe-L[c]',        # Phenylalanine,    ChEBI 17295
    'P': 'pro-L[c]',        # Proline,          ChEBI 17203
    'S': 'ser-L[c]',        # Serine,           ChEBI 17115
    'T': 'thr-L[c]',        # Threonine,        ChEBI 16857
    'W': 'trp-L[c]',        # Tryptophan,       ChEBI 16828
    'Y': 'tyr-L[c]',        # Tyrosine,         ChEBI 17895
    'V': 'val-L[c]',        # Valine,           ChEBI 16414
    'h2o': 'h2o[c]',        # H2O
    'adp': 'adp[c]',        # ADP
    'Pi': 'pi[c]',          # Phosphate
    'h': 'h[c]',            # Hydrogen [Proton]
    'PPi': 'ppi[c]',        # Pyrophosphate
}
# Nucleotide dictionary with molecular weights
# Source: ChEBI https://www.ebi.ac.uk/chebi/
ntpsDict = {
    'atp': 507.181,     # ATP,              ChEBI 15422
    'gtp': 483.15644,   # GTP,              ChEBI 17677
    'ctp': 523.18062,   # CTP,              ChEBI 15996
    'utp': 484.14116,   # UTP,              ChEBI 15713 (TTP is
    psuedo for UTP in viral genome)
}
# Amino Acids dictionary with molecular weights
# Source: ChEBI https://www.ebi.ac.uk/chebi/
aaDict = {
    'A': 89.09322,      # Alanine,          ChEBI 16977
    'R': 174.201,       # Arginine,         ChEBI 16467
    'N': 132.118,       # Asparagine,       ChEBI 17196
    'D': 133.1027,      # Aspartate,        ChEBI 17053
    'C': 121.158,       # Cysteine,         ChEBI 17561
    'Q': 146.14458,     # Glutamine,        ChEBI 18050
    'E': 147.1293,      # Glutamate,        ChEBI 16015
```

```
52      'G': 75.06664,      # Glycine,          ChEBI 15428
53      'H': 155.15468,     # Histidine,        ChEBI 15971
54      'I': 131.17296,     # Isoleucine,       ChEBI 17191
55      'L': 131.17296,     # Leucine,          ChEBI 15603
56      'K': 146.18764,     # Lysine,           ChEBI 18019
57      'M': 149.21238,     # Methionine,       ChEBI 16643
58      'F': 165.18918,     # Phenylalanine,    ChEBI 17295
59      'P': 115.1305,      # Proline,          ChEBI 17203
60      'S': 105.09262,     # Serine,           ChEBI 17115
61      'T': 119.1192,      # Threonine,        ChEBI 16857
62      'W': 204.22526,     # Tryptophan,       ChEBI 16828
63      'Y': 181.18858,     # Tyrosine,         ChEBI 17895
64      'V': 117.14638,     # Valine,           ChEBI 16414
65  }
66  # Misc. dictionary with molecular weights
67  # Source: ChEBI https://www.ebi.ac.uk/chebi/
68  miscDict = {
69      'PPi': 173.94332,   # Pyrophosphate,    ChEBI 18361
70  }
71  # Nucleotides List
72  ntpsMets    = list(ntpsDict.keys())
73  # Amino Acids List
74  aaMets      = list(aaDict.keys())
75  # Avogadro's Number
76  N_A         = 6.0221409e+23
77  # Energy requirement coefficients
78  # Source: Haynie, D. (2008). Biological Thermodynamics (2nd ed.).
        Cambridge: Cambridge University Press.
79  k_atp       = 4
80  k_ppi       = 1
```

Listing 3: Viranet Sub-Package: info.py

A.4 tools.py

```
# VIRANET TOOLS #
#######
# objSto
# Strips stoichiometric coefficients from the objectives reactions
# usage
# Inputs:
# HVM                Integrated host-virus model
# HostRxn            Host objective reaction, either:
#                    - Index value of reaction in Model.reactions [int]
#                    - Reaction ID of the host-objective reaction [str]
#
# Outputs:
# hostSto            Stoichiometric coefficients with metabolites for host objective
# virusSto           Stoichiometric coefficients with metabolites for virus objective
#

def objSto(HVM,HostRxn):
    "Differential usage of amino acids and nucleotides"
    # [1] Initial Setup
    # Function Dependencies
    import pandas   as pd
    import numpy    as np
    # Identify the host objective reaction
    try:
        intTest = int(HostRxn)
        hostIdx = HostRxn
    except:
        for ii in range(len(HVM.reactions)):
            if HostRxn in str(HVM.reactions[ii]):
                hostIdx = ii
    virusIdx    = len(HVM.reactions) - 1    # ViraNet(c) appends virus reaction to end in genHVM.py
    objIdx      = [hostIdx,virusIdx]
    hostID      = HVM.reactions[hostIdx].id
    virusID     = HVM.reactions[virusIdx].id
    # Condition: ensure virus reaction is virus objective
    if '_prodrxn_VN' not in virusID:
        raise ValueError('Unsupported objective, unable to analyse: refer to README')
    # Convert model into an array
    m       = HVM.to_array_based_model()
    # Create data frame
    mFrame  = pd.DataFrame(
        data    = m.S.todense(),
        columns = m.reactions.list_attr("id"),
        index   = m.metabolites.list_attr("id")
    )
    # [2] Stoichiometric Data
```

```
    # Strip the host and virus objective function stoichiometric
    coefficients
    hostS   = mFrame[hostID]
    virusS  = mFrame[virusID]
    # Record only non zero values
    hostSto     = hostS.loc[~(hostS==0)]
    virusSto    = virusS.loc[~(virusS==0)]
    # [3] Create output
    hostSto     = pd.DataFrame(hostSto)
    virusSto    = pd.DataFrame(virusSto)
    return (hostSto,virusSto)
#######

#######
# boundCheck
# Checks a model for any arbitrarily large bounds, removes these
# and replaces with infinite bounds
# Work based upon Kelk et al 2012
# Kelk, S. M., Olivier, B. G., Stougie, L., & Bruggeman, F. J. (2012)
    . Optimal flux spaces of genome-scale stoichiometric models are
    determined by a few subnetworks. Scientific Reports, 2, 580. http
    ://doi.org/10.1038/srep00580
# Inputs:
# Model         User-supplied model file    [.mat,.xml]
#
# Outputs:
# infModel      Model with bound corrections applied
def boundCheck(Model):
    "Reaction bound checker"
    # [1] Initial Setup
    # Function Dependencies
    import numpy    as np
    # Create pointer
    altModel = Model
    # [2] Identify and correct arbitrarily large reaction bounds
    for ii in range(len(altModel.reactions)):
        # Temporarily record the lower and upper bounds
        tmpLb   = altModel.reactions[ii].lower_bound
        tmpUb   = altModel.reactions[ii].upper_bound
        # Conditional statement
        if tmpLb <= -1000:
            altLb   = -np.inf
            altModel.reactions[ii].lower_bound = altLb
        if tmpUb >= 1000:
            altUb   = np.inf
            altModel.reactions[ii].upper_bound = altUb
    # [3] Output model
    return altModel
#######
# rangeCalculator
# Calculates , using FVA results for host and virus, the flux range
    to use in the
# host-derived enforcement analysis
# Inputs:
```

```
# hostIdx           Index (model.reactions) for the host-objective reaction
# virusIdx          Index (model.reactions) for the virus-objective reaction
# Optional Inputs
# solver            Declare solver to use for cobrapy: default is cglpk
# Outputs
# enfVirus          Vector of virus optima values with additional host-constraint
# maxEnfBound       Maximum enf bound
# minEnfBound       Minimum enf bound
def rangeCalculator(HVM, hostIdx, virusIdx, solver):
    "Enforcement bound creator"
    # [1] Initial Setup
    # Function Dependencies
    import cobra
    import numpy    as np
    import pandas   as pd
    # Create the virus optima vector
    enfVirus    = np.zeros((len(HVM.reactions),1))
    maxEnfBound = np.zeros((len(HVM.reactions),1))
    minEnfBound = np.zeros((len(HVM.reactions),1))
    # [2] Perform FVA for each objective
    # Objective reactions
    hostObj     = HVM.reactions[hostIdx]
    virusObj    = HVM.reactions[virusIdx]
    # Host Optimisation
    HVM.change_objective(hostObj)
    # Ensure no flux can go through the virus reaction
    # Store the bounds
    virusLb     = HVM.reactions[virusIdx].lower_bound
    virusUb     = HVM.reactions[virusIdx].upper_bound
    HVM.reactions[virusIdx].lower_bound = 0
    HVM.reactions[virusIdx].upper_bound = 0
    # FVA
    varHost     = cobra.flux_analysis.flux_variability_analysis(HVM, solver=solver)
    # Return virus objective bounds
    HVM.reactions[virusIdx].lower_bound = virusLb
    HVM.reactions[virusIdx].upper_bound = virusUb
    # Virus Optimisation
    HVM.change_objective(virusObj)
    # Ensure no flux can go through the host reaction
    # Store the bounds
    hostLb      = HVM.reactions[hostIdx].lower_bound
    hostUb      = HVM.reactions[hostIdx].upper_bound
    HVM.reactions[hostIdx].lower_bound = 0
    HVM.reactions[hostIdx].upper_bound = 0
    # FVA
    varVirus    = cobra.flux_analysis.flux_variability_analysis(HVM, solver=solver)
    # Return host objective bounds
    HVM.reactions[hostIdx].lower_bound = hostLb
```

```
144     HVM.reactions[hostIdx].upper_bound = hostUb
145     # Create data frames
146     hostFVA      = pd.DataFrame.from_dict(varHost)
147     hostFVA      = hostFVA.transpose()
148     virusFVA     = pd.DataFrame.from_dict(varVirus)
149     virusFVA     = virusFVA.transpose()
150     # [3] Condition statements to determine the calculation and FVA
    steps
151     # Initiate loop
152     for ii in range(len(HVM.reactions)):
153         # Create temporary host and virus max|min variables
154         hostMax      = varHost[HVM.reactions[ii].id]['maximum']
155         hostMin      = varHost[HVM.reactions[ii].id]['minimum']
156         virusMax     = varVirus[HVM.reactions[ii].id]['maximum']
157         virusMin     = varVirus[HVM.reactions[ii].id]['minimum']
158         # Record the upper and lower bounds
159         tmpLb        = HVM.reactions[ii].lower_bound
160         tmpUb        = HVM.reactions[ii].upper_bound
161         # Conditional: H+ > V+ && H- < V-
162         if (hostMax > virusMax) and (hostMin < virusMin):
163             ########
164             # Calculation of bounds for condition [1]
165             enfMax1 = hostMax
166             enfMin1 = (hostMax - ((hostMax - virusMax) / 2))
167             # Apply bounds to reaction
168             HVM.reactions[ii].lower_bound    = enfMin1
169             HVM.reactions[ii].upper_bound    = enfMax1
170             # Zero-bound host
171             HVM.reactions[hostIdx].lower_bound = 0
172             HVM.reactions[hostIdx].upper_bound = 0
173             # Optimize for virus
174             HVM.change_objective(virusObj)
175             sol = HVM.optimize(objective_sense='maximize',solver=
    solver)
176             # Record the optima
177             zMax     = sol.f
178             # Return host bounds to original
179             HVM.reactions[hostIdx].lower_bound = hostLb
180             HVM.reactions[hostIdx].upper_bound = hostUb
181             # Return reaction bounds to original
182             HVM.reactions[ii].lower_bound    = tmpLb
183             HVM.reactions[ii].upper_bound    = tmpUb
184             # Calculation of bounds for condition [2]
185             enfMin2 = hostMin
186             enfMax2 = (hostMin - ((hostMin - virusMin) / 2))
187             # Apply bounds to reaction
188             HVM.reactions[ii].lower_bound    = enfMin2
189             HVM.reactions[ii].upper_bound    = enfMax2
190             # Zero-bound host
191             HVM.reactions[hostIdx].lower_bound = 0
192             HVM.reactions[hostIdx].upper_bound = 0
193             # Optimize for virus
194             HVM.change_objective(virusObj)
195             sol = HVM.optimize(objective_sense='maximize',solver=
```

```
    solver)
196             # Record the optima
197             zMin     = sol.f
198             # Return host bounds to original
199             HVM.reactions[hostIdx].lower_bound = hostLb
200             HVM.reactions[hostIdx].upper_bound = hostUb
201             # Return reaction bounds to original
202             HVM.reactions[ii].lower_bound    = tmpLb
203             HVM.reactions[ii].upper_bound    = tmpUb
204             # COMPARISON #
205             # Compare zMax and zMin to find which is smallest
206             if zMax < zMin:
207                 enfVirus[ii] = zMax
208                 # Record the bound
209                 maxEnfBound[ii] = enfMax1
210                 minEnfBound[ii] = enfMin1
211             elif zMin < zMax:
212                 enfVirus[ii] = zMin
213                 # Record the bound
214                 maxEnfBound[ii] = enfMax2
215                 minEnfBound[ii] = enfMin2
216             else:
217                 enfVirus[ii] = zMax
218                 # Record the bound
219                 maxEnfBound[ii] = enfMax1
220                 minEnfBound[ii] = enfMin1
221         # Conditional: H+ > V+
222         else:
223             if hostMax > virusMax:
224                 #######
225                 # Calculation of bounds
226                 enfMax  = hostMax
227                 enfMin  = (hostMax - ((hostMax - virusMax) / 2))
228                 # Apply bounds to reaction
229                 HVM.reactions[ii].lower_bound    = enfMin
230                 HVM.reactions[ii].upper_bound    = enfMax
231                 # Zero-bound host
232                 HVM.reactions[hostIdx].lower_bound = 0
233                 HVM.reactions[hostIdx].upper_bound = 0
234                 # Optimize for virus
235                 HVM.change_objective(virusObj)
236                 sol = HVM.optimize(objective_sense='maximize',solver=
    solver)
237                 # Record the optima
238                 enfVirus[ii]     = sol.f
239                 # Return host bounds to original
240                 HVM.reactions[hostIdx].lower_bound = hostLb
241                 HVM.reactions[hostIdx].upper_bound = hostUb
242                 # Return reaction bounds to original
243                 HVM.reactions[ii].lower_bound    = tmpLb
244                 HVM.reactions[ii].upper_bound    = tmpUb
245                 # Record the bound
246                 maxEnfBound[ii] = enfMax
247                 minEnfBound[ii] = enfMin
```

```
248                 #######
249             else:
250                 # Conditional: H- < V-
251                 if hostMin < virusMin:
252                     #######
253                     enfMin  = hostMin
254                     enfMax  = (hostMin - ((hostMin - virusMin) / 2))
255                     # Apply bounds to reaction
256                     HVM.reactions[ii].lower_bound   = enfMin
257                     HVM.reactions[ii].upper_bound   = enfMax
258                     # Zero-bound host
259                     HVM.reactions[hostIdx].lower_bound = 0
260                     HVM.reactions[hostIdx].upper_bound = 0
261                     # Optimize for virus
262                     HVM.change_objective(virusObj)
263                     sol = HVM.optimize(objective_sense='maximize',
    solver=solver)
264                     # Record the optima
265                     enfVirus[ii]    = sol.f
266                     # Return host bounds to original
267                     HVM.reactions[hostIdx].lower_bound = hostLb
268                     HVM.reactions[hostIdx].upper_bound = hostUb
269                     # Return reaction bounds to original
270                     HVM.reactions[ii].lower_bound   = tmpLb
271                     HVM.reactions[ii].upper_bound   = tmpUb
272                     # Record the bound
273                     maxEnfBound[ii] = enfMax
274                     minEnfBound[ii] = enfMin
275                     #######
276                 else:
277                     #######
278                     enfMax  = hostMax
279                     enfMin  = hostMin
280                     # Apply bounds to reaction
281                     HVM.reactions[ii].lower_bound   = enfMin
282                     HVM.reactions[ii].upper_bound   = enfMax
283                     # Zero-bound host
284                     HVM.reactions[hostIdx].lower_bound = 0
285                     HVM.reactions[hostIdx].upper_bound = 0
286                     # Optimize for virus
287                     HVM.change_objective(virusObj)
288                     sol = HVM.optimize(objective_sense='maximize',
    solver=solver)
289                     # Record the optima
290                     enfVirus[ii]    = sol.f
291                     # Return host bounds to original
292                     HVM.reactions[hostIdx].lower_bound = hostLb
293                     HVM.reactions[hostIdx].upper_bound = hostUb
294                     # Return reaction bounds to original
295                     HVM.reactions[ii].lower_bound   = tmpLb
296                     HVM.reactions[ii].upper_bound   = tmpUb
297                     # Record the bound
298                     maxEnfBound[ii] = enfMax
299                     minEnfBound[ii] = enfMin
```

```
300                         #######
301     # [4] Output
302     return (enfVirus,maxEnfBound,minEnfBound)
303 #######
```

Listing 4: Viranet Sub-Package: tools.py

A.5 main.py

```
#######
# main.py performs analysis on a user-defined host model, and a user-
# supplied virus genome file (from NCBI > Genbank)
# A host-virus integrated model is created and the following analysis performed:
# - Comparison of flux distribution
# - Analysis of the effect of single-reaction knockouts on virus optima
# - Analysis of the effect of single-reaction host-derived flux enforcement on
#   virus optima
#######
# Setup of workspace
# Python Dependencies
import  cobra
import  numpy       as np
import  pandas      as pd
import  os
import  re
import  csv
import  time
# Intra-package dependencies
from    viranet     import generation
from    viranet     import analysis
from    viranet     import info
from    viranet     import tools

#######
# singleAnalysis
# Performs analysis on a integrated host-virus model, for a user-defined
# metabolic model and user-defined virus
# Inputs:
# Model             User-supplied model file for desired host   [.mat,.xml]
# VirusGB           User-supplied GenBank file (NCBI) for desired virus [.gb, .txt]
# HostRxn           Host objective reaction, either:
#                   - Index value of reaction in Model.reactions [int]
#                   - Reaction ID of the host-objective reaction [str]
# Optional Inputs:
# solver            User-specified solver (default = CGLPK)
# preanalysis       User-specified pre-analysis of model (default = False)
#                   - Checking for arbitrarily large (> 1000) reaction flux bounds
# usefva            Use the Host FBA result (False) or use the FVA range as
#                   the additional constraint (True)
#
```

```
# Outputs:
# ViraNet_Analysis  Full analysis of the HVM, written to .csv files

def singleAnalysis(Model,VirusGB,HostRxn,solver="cglpk",fbaType="fba",preanalysis=False,usefva=False,userange=False):
    "Analysis of Host-Virus model"
    # Generate the neccessary virus objective function and integrated host-virus model
    # require for the analysis
    # [1] VBOF Generation
    virusReaction = generation.VBOF(VirusGB)
    print("[1] VBOF Generated")

    # [2] Intergrated Host-Virus Model (HVM)
    (virusModel) = generation.HVM(Model,virusReaction)
    print("[2] HVM Generated")

    # OPTIONAL STEPS #
    if preanalysis == True:
        # [I] Checking model for arbitrarily large bounds
        virusModel = tools.boundCheck(virusModel)
        print("[2A] HVM Reaction Bounds Checked")

    # [3] Host-Virus Optimisation
    # Optimise the original model
    (objIdx,hostF,hostX,virusF,virusX) = analysis.optimise(virusModel,HostRxn,solver,fbaType)
    print("[3] HVM Optimisation Complete")

    # [4] Host-Virus Differential Analysis
    (rel_dif, abs_dif) = analysis.differential(virusModel,HostRxn)
    print("[4] HVM Differential Analysis Complete")

    # [5] Host-Virus Comparison
    (hvmComp,hvmStat,hostXd,virusXd) = analysis.compare(objIdx,hostX,virusX)
    print("[5] HVM Comparison Complete")

    # [6] Host-Virus Variability Analysis
    (hostFVA,virusFVA) = analysis.variability(virusModel,HostRxn,solver)
    print("[6] HVM Variability Analysis Complete")

    # [7] Knockout Analysis
    (koVirus)   = analysis.knockout(virusModel,HostRxn,solver)
    print("[7] HVM Knockout Analysis Complete")

    # [8] Enforcement Analysis
    # Conditional on userange
    if userange == False:
        (enfVirus, _, _) = analysis.enforce(virusModel,hostX,HostRxn,solver,usefva,userange)
    elif userange == True:
        (enfVirus,maxEnfBound,minEnfBound) = analysis.enforce(
```

```
    virusModel,hostX,HostRxn,solver,usefva,userange)
90  print("[8] HVM Host-Derived Enforcement Analysis Complete")
91
92  # [9] Function output: setup
93  # Identify the host objective reaction
94  try:
95      intTest = int(HostRxn)
96      hostIdx = HostRxn
97  except:
98      for ii in range(len(virusModel.reactions)):
99          if HostRxn in str(virusModel.reactions[ii]):
100             hostIdx = ii
101 # Record the other reactions
102 virusIdx    = len(virusModel.reactions) - 1    # ViraNet(c)
    appends virus reaction to end in genHVM.py
103 objIdx      = [hostIdx,virusIdx]
104 hostID      = virusModel.reactions[hostIdx].id
105 virusID     = virusModel.reactions[virusIdx].id
106 # Condition: ensure virus reaction is virus objective
107 if '_prodrxn_VN' not in virusID:
108     raise ValueError('Unsupported objective, unable to analyse:
    refer to README')
109 # Create array-model and obtain neccessary information
110 m       = virusModel.to_array_based_model()
111 mRxns   = m.reactions.list_attr("id")
112 # Remove the objective reactions
113 if hostID in mRxns:
114     mRxns.remove(hostID)
115 if virusID in mRxns:
116     mRxns.remove(virusID)
117 # Obtain subsystem information
118 subSystems = []
119 for ii in range(len(virusModel.reactions)):
120     if not(ii in objIdx):
121         subSystems.append(virusModel.reactions[ii].subsystem)
122
123 # [10] singleAnalysis Output Generation
124 # Virus name and date
125 date        = (time.strftime("%d_%m_%Y"))
126 idVirus     = virusID.replace("_prodrxn_VN","")
127 idVirus     = "_" + idVirus + "_" + date
128
129 # OUTPUT [1]: Host-Virus Optimisation Comparisons
130 # Store as list
131 hostFlux        = list(hvmComp[:,0])
132 virusFlux       = list(hvmComp[:,1])
133 hostSolX        = list(hostXd)
134 virusSolX       = list(virusXd)
135 # Store in dataframe
136 hvmComparison   = pd.DataFrame([mRxns,subSystems,hostSolX,
    virusSolX,hostFlux,virusFlux])
137 hvmComparison   = hvmComparison.transpose()
138 hvmComparison.columns = ['Reaction','Subsystem','Host_Flux_mmol/
    gDW/h-1','Virus_Flux__mmol/gDW/h-1','Host_Flux_%','Virus_Flux_%']
```

```
139     title = "ViraNet_Comparison" + idVirus + ".csv"
140     # Write to csv
141     hvmComparison.to_csv(title,index=False)
142 
143     # OUTPUT [2]: Host-Virus Optimisation Comparison Stats
144     # Store as list
145     compStats       = list(hvmStat)
146     nameStats       = ['Upregulated','Downregulated','Activated','
    Inactivated','Reversed']
147     # Store in dataframe
148     hvmComparisonStats  = pd.DataFrame([nameStats,compStats])
149     hvmComparisonStats  = hvmComparisonStats.transpose()
150     hvmComparisonStats.columns  = ['Regulation_state','Num_Rxns']
151     title = "ViraNet_ComparisonStats" + idVirus + ".csv"
152     # Write to csv
153     hvmComparisonStats.to_csv(title,index=False)
154 
155     # OUTPUT[3]: Host-virus variability analysis using FVA
156     # Create title host
157     title = "ViraNet_HostFVA" + idVirus + ".csv"
158     # Write to csv
159     hostFVA.to_csv(title,index=True)
160     # Create title virus
161     title = "ViraNet_VirusFVA" + idVirus + ".csv"
162     # Write to csv
163     virusFVA.to_csv(title,index=True)
164 
165     # OUTPUT[4]: Host-virus differential usage of amino acids and
    nucleotides
166     # Create lists for the relative and absolute measurements
167     relativeDiff    = list(rel_dif.values())
168     absoluteDiff    = list(abs_dif.values())
169     orderedMets     = list(rel_dif.keys())
170     # Store in dataframe
171     hvmDifferential     = pd.DataFrame([orderedMets,relativeDiff,
    absoluteDiff])
172     hvmDifferential     = hvmDifferential.transpose()
173     hvmDifferential.columns     = ['Obj_Met','Relative_d','Absolute_d
    ']
174     title = "ViraNet_Differential_Usage" + idVirus + ".csv"
175     # Write to csv
176     hvmDifferential.to_csv(title,index=False)
177 
178     # OUTPUT [5]: Virus optima and effects of knockout and host-
    derived enforcement
179     # Store as list whilst removing objective reactions
180     resultKO        = list(np.delete(koVirus,objIdx))
181     resultEF        = list(np.delete(enfVirus,objIdx))
182     if userange == True:
183         resultMaxEnfBound   = list(np.delete(maxEnfBound,objIdx))
184         resultMinEnfBound   = list(np.delete(minEnfBound,objIdx))
185     # Convert to percentage of wild-type [unconstrained] virus optima
186     for ii in range(len(resultKO)):
187         if np.isnan(resultKO[ii]) == False:
```

```
188             resultKO[ii]     = (resultKO[ii] / virusF) * 100
189         elif np.isnan(resultKO[ii]) == True:
190             resultKO[ii]     = 0
191         elif resultKO[ii] is None:
192             resultKO[ii]     = 0
193         if np.isnan(resultEF[ii]) == False:
194             resultEF[ii]     = (resultEF[ii] / virusF) * 100
195         elif np.isnan(resultEF[ii]) == True:
196             resultEF[ii]     = 0
197         elif resultEF[ii] is None:
198             resultEF[ii]     = 0
199     # Store in dataframe
200     if userange == False:
201         hvmKoEnf         = pd.DataFrame([mRxns,subSystems,resultKO,
    resultEF])
202         hvmKoEnf         = hvmKoEnf.transpose()
203         hvmKoEnf.columns     = ['Reaction','Subsystem','
    Knockout_Optima(%_WT)','Enforcement_Optima(%_WT)']
204         title = "ViraNet_Knockout_Enforcement" + idVirus + ".csv"
205         # Write to csv
206         hvmKoEnf.to_csv(title,index=False)
207         #return (hvmComp, hvmStat, koVirus, enfVirus)
208     elif userange == True:
209         hvmKoEnf         = pd.DataFrame([mRxns,subSystems,resultKO,
    resultEF,resultMaxEnfBound,resultMinEnfBound])
210         hvmKoEnf         = hvmKoEnf.transpose()
211         hvmKoEnf.columns     = ['Reaction','Subsystem','
    Knockout_Optima(%_WT)','Enforcement_Optima(%_WT)','
    MaximumBound_Enf','MinimumBound_Enf']
212         title = "ViraNet_Knockout_Enforcement" + idVirus + ".csv"
213         # Write to csv
214         hvmKoEnf.to_csv(title,index=False)
215         #return (hvmComp, hvmStat, koVirus, enfVirus)
216     print("ViraNet Analysis Complete")
217 #######
```

Listing 5: Viranet Sub-Package: main.py

B HOST BIOMASS OBJECTIVE FUNCTION COMPOSITION

Table B1. Biomass objective function stoichiometric coefficients and associated metabolites for the [host] human macrophage (iAB-AMO-1410) biomass maintenance objective function.

Model Metabolite ID	Metabolite Name	Macromolecule	Stoichiometric Coefficients, S [mmol/gDW]
adp[c]	ADP	Energy Requirement	25.17352552
ala-L[c]	L-Alanine	Amino Acids	-0.396559456
alpa_hs[c]	lysophosphatidic acid (homo sapiens)	Other	-0.011499127
amp[c]	AMP	RNA Nucleotide	-0.048664064
arg-L[c]	L-Arginine	Amino Acids	-0.325724532
asn-L[c]	L-Asparagine	Amino Acids	-0.215407845
asp-L[c]	L-Aspartate	Amino Acids	-0.282759085
atp[c]	ATP	Energy Requirement	-25.17352552
chsterol[c]	Cholesterol	Other	-0.020930954
cmp[c]	CMP	RNA Nucleotide	-0.042373167
cys-L[c]	L-Cysteine	Amino Acids	-0.127154496
dag_hs[c]	diacylglycerol (homo sapiens)	Other	-0.0036682
damp[c]	dAMP	DNA Nucleotide	-0.021495345
dcmp[c]	dCMP	DNA Nucleotide	-0.014937443
dgmp[c]	dGMP	DNA Nucleotide	-0.014937443
dtmp[c]	dTMP	DNA Nucleotide	-0.021495345

Continued on next page

Table B1 – continued from previous page

Model Metabolite ID	Metabolite Name	Macromolecule	Stoichiometric Coefficients, S [mmol/gDW]
h2o[c]	H2O	Energy Requirement	-25.17352552
gln-L[c]	L-Glutamine	Amino Acids	-0.280436629
glu-L[c]	L-Glutamate	Amino Acids	-0.424428935
gly[c]	Glycine	Amino Acids	-0.366948135
glygn2[c]	glycogen, structure 2 (glycogenin-1,6-{7[1,4-Glc], 4[1,4-Glc]})	Other	-0.528027894
gmp[c]	GMP	RNA Nucleotide	-0.043710887
h[c]	H+	Energy Requirement	25.17352552
hdca[c]	Hexadecanoate (n-C16:0)	Other	-0.004850777
hdcea[c]	Hexadecenoate (n-C16:1)	Other	-0.001222285
his-L[c]	L-Histidine	Amino Acids	-0.153862747
ile-L[c]	L-Isoleucine	Amino Acids	-0.25953452
leu-L[c]	L-Leucine	Amino Acids	-0.580614138
lys-L[c]	L-Lysine	Amino Acids	-0.351852168
met-L[c]	L-Methionine	Amino Acids	-0.126573882
ocdca[c]	octadecanoate (n-C18:0)	Other	-0.004736708

Continued on next page

Table B1 – continued from previous page

Model Metabolite ID	Metabolite Name	Macromolecule	Stoichiometric Coefficients, S [mmol/gDW]
ocdcea[c]	octadecenoate (n-C18:1)	Other	-0.003853116
pail_hs[c]	phosphatidylinositol (homo sapiens)	Other	-0.003741686
pchol_hs[c]	Phosphatidylcholine (homo sapiens)	Other	-0.031527146
pe_hs[c]	phosphatidylethanolamine (homo sapiens)	Other	-0.021107135
pglyc_hs[c]	phosphatidylglycerol (homo sapiens)	Other	-0.008918017
phe-L[c]	L-Phenylalanine	Amino Acids	-0.214246617
pi[c]	Phosphate	Energy Requirement	25.17352552
pro-L[c]	L-Proline	Amino Acids	-0.346626641
ps_hs[c]	phosphatidylserine (homo sapiens)	Other	-0.001024655
ser-L[c]	L-Serine	Amino Acids	-0.476684207
sphmyln_hs[c]	sphingomyelin (homo sapiens)	Other	-0.007049706
tag_hs[c]	triacylglycerol (homo sapiens)	Other	-0.002742439

Continued on next page

Table B1 – continued from previous page

Model Metabolite ID	Metabolite Name	Macromolecule	Stoichiometric Coefficients, S [mmol/gDW]
thr-L[c]	L-Threonine	Amino Acids	-0.303661194
trp-L[c]	L-Tryptophan	Amino Acids	-0.069673697
ttdca[c]	tetradecanoate (n-C14:0)	Other	-0.00136164
tyr-L[c]	L-Tyrosine	Amino Acids	-0.156185203
ump[c]	UMP	RNA Nucleotide	-0.04602478
val-L[c]	L-Valine	Amino Acids	-0.347207255

C VIRUS BIOMASS OBJECTIVE FUNCTION COMPOSITIONS

Table C1. Biomass objective function stoichiometric coefficients and associated metabolites for the Chikungunya (CHIKV) virus biomass objective functions. ∗ ATP includes both the nucleotide and energy requirement fractions as a single value.

Model Metabolite ID	Metabolite Name	Macromolecule	Stoichiometric Coefficients, S [mmol/gDW]
adp[c]	ADP	Energy Requirement	23.82348236
ala-L[c]	L-Alanine	Amino Acids	-0.491958902
arg-L[c]	L-Arginine	Amino Acids	-0.272990314
asn-L[c]	L-Asparagine	Amino Acids	-0.257614748
asp-L[c]	L-Aspartate	Amino Acids	-0.211191986
atp[c]*	ATP*	Nucleotide	-23.94009192
ctp[c]	CTP	Nucleotide	-0.116806932
cys-L[c]	L-Cysteine	Amino Acids	-0.233377013
gln-L[c]	L-Glutamine	Amino Acids	-0.257476585
glu-L[c]	L-Glutamate	Amino Acids	-0.29201733
gly[c]	Glycine	Amino Acids	-0.362677826
gtp[c]	GTP	Nucleotide	-0.116806932
h[c]	H+	Energy Requirement	23.82348236
h2o[c]	H2O	Energy Requirement	-23.82348236
his-L[c]	L-Histidine	Amino Acids	-0.200138947
ile-L[c]	L-Isoleucine	Amino Acids	-0.291069927
leu-L[c]	L-Leucine	Amino Acids	-0.392678931
lys-L[c]	L-Lysine	Amino Acids	-0.376948088

Continued on next page

Table C1 – continued from previous page

Model Metabolite ID	Metabolite Name	Macromolecule	Stoichiometric Coefficients, S [mmol/gDW]
met-L[c]	L-Methionine	Amino Acids	-0.148051503
phe-L[c]	L-Phenylalanine	Amino Acids	-0.162637567
pi[c]	Phosphate	Energy Requirement	23.82348236
ppi[c]	Diphosphate	Energy Requirement	0.466793502
pro-L[c]	L-Proline	Amino Acids	-0.433752811
ser-L[c]	L-Serine	Amino Acids	-0.349098379
thr-L[c]	L-Threonine	Amino Acids	-0.439239856
trp-L[c]	L-Tryptophan	Amino Acids	-0.071548687
tyr-L[c]	L-Tyrosine	Amino Acids	-0.233771765
utp[c]	UTP	Nucleotide	-0.116609556
val-L[c]	L-Valine	Amino Acids	-0.482386181

Table C2. Biomass objective function stoichiometric coefficients and associated metabolites for the Dengue (DENV) virus biomass objective functions. ∗ ATP includes both the nucleotide and energy requirement fractions as a single value.

Model Metabolite ID	Metabolite Name	Macromolecule	Stoichiometric Coefficients, S [mmol/gDW]
adp[c]	ADP	Energy Requirement	19.49420757
ala-L[c]	L-Alanine	Amino Acids	-0.241248287
arg-L[c]	L-Arginine	Amino Acids	-0.264837466
asn-L[c]	L-Asparagine	Amino Acids	-0.182944901
asp-L[c]	L-Aspartate	Amino Acids	-0.139783913
atp[c]*	ATP*	Nucleotide	-19.69370267
ctp[c]	CTP	Nucleotide	-0.168695251
cys-L[c]	L-Cysteine	Amino Acids	-0.112589401
gln-L[c]	L-Glutamine	Amino Acids	-0.169450654
glu-L[c]	L-Glutamate	Amino Acids	-0.303294352
gly[c]	Glycine	Amino Acids	-0.445790849
gtp[c]	GTP	Nucleotide	-0.168695251
h[c]	H+	Energy Requirement	19.49420757
h2o[c]	H2O	Energy Requirement	-19.49420757
his-L[c]	L-Histidine	Amino Acids	-0.119079001
ile-L[c]	L-Isoleucine	Amino Acids	-0.321011989
leu-L[c]	L-Leucine	Amino Acids	-0.434871841
lys-L[c]	L-Lysine	Amino Acids	-0.327055214

Continued on next page

Table C2 – continued from previous page

Model Metabolite ID	Metabolite Name	Macromolecule	Stoichiometric Coefficients, S [mmol/gDW]
met-L[c]	L-Methionine	Amino Acids	-0.244235563
phe-L[c]	L-Phenylalanine	Amino Acids	-0.18167445
pi[c]	Phosphate	Energy Requirement	19.49420757
ppi[c]	Diphosphate	Energy Requirement	0.736312022
pro-L[c]	L-Proline	Amino Acids	-0.189400164
ser-L[c]	L-Serine	Amino Acids	-0.264734457
thr-L[c]	L-Threonine	Amino Acids	-0.445344475
trp-L[c]	L-Tryptophan	Amino Acids	-0.101430037
tyr-L[c]	L-Tyrosine	Amino Acids	-0.070080808
utp[c]	UTP	Nucleotide	-0.199495096
val-L[c]	L-Valine	Amino Acids	-0.320908979

Table C3. Biomass objective function stoichiometric coefficients and associated metabolites for the Zika (ZIKV) virus biomass objective functions. ∗ ATP includes both the nucleotide and energy requirement fractions as a single value.

Model Metabolite ID	Metabolite Name	Macromolecule	Stoichiometric Coefficients, S [mmol/gDW]
adp[c]	ADP	Energy Requirement	19.76542883
ala-L[c]	L-Alanine	Amino Acids	-0.388789243
arg-L[c]	L-Arginine	Amino Acids	-0.270454656
asn-L[c]	L-Asparagine	Amino Acids	-0.144194657
asp-L[c]	L-Aspartate	Amino Acids	-0.22539596
atp[c]*	ATP*	Nucleotide	-19.94634684
ctp[c]	CTP	Nucleotide	-0.18781858
cys-L[c]	L-Cysteine	Amino Acids	-0.118368748
gln-L[c]	L-Glutamine	Amino Acids	-0.106548954
glu-L[c]	L-Glutamate	Amino Acids	-0.239948655
gly[c]	Glycine	Amino Acids	-0.475251379
gtp[c]	GTP	Nucleotide	-0.18781858
h[c]	H+	Energy Requirement	19.76542883
h2o[c]	H2O	Energy Requirement	-19.76542883
his-L[c]	L-Histidine	Amino Acids	-0.14316982
ile-L[c]	L-Isoleucine	Amino Acids	-0.275032264
leu-L[c]	L-Leucine	Amino Acids	-0.463192456
lys-L[c]	L-Lysine	Amino Acids	-0.312165548

Continued on next page

Table C3 – continued from previous page

Model Metabolite ID	Metabolite Name	Macromolecule	Stoichiometric Coefficients, S [mmol/gDW]
met-L[c]	L-Methionine	Amino Acids	-0.181772038
phe-L[c]	L-Phenylalanine	Amino Acids	-0.15598029
pi[c]	Phosphate	Energy Requirement	19.76542883
ppi[c]	Diphosphate	Energy Requirement	0.73740485
pro-L[c]	L-Proline	Amino Acids	-0.163837379
ser-L[c]	L-Serine	Amino Acids	-0.330885916
thr-L[c]	L-Threonine	Amino Acids	-0.368497457
trp-L[c]	L-Tryptophan	Amino Acids	-0.10098067
tyr-L[c]	L-Tyrosine	Amino Acids	-0.119154457
utp[c]	UTP	Nucleotide	-0.180918006
val-L[c]	L-Valine	Amino Acids	-0.363919849

D RESULTS OF SINGLE-REACTION ANALYSES

Table D1. Results for the single-reaction knockout analysis. Model reaction identifiers are listed. The resulting virus optima, for Chikungunya (CHIKV), Dengue (DENV) and Zika (ZIKV), are shown as a percentage of the original model (without the additional constraints imposed by the analyses), identified as normalised.

Reaction ID	**CHIKV Optima (Normalised)**	**DENV Optima (Normalised)**	**ZIKV Optima (Normalised)**
ADSL1	0	0	0
ADSL2	0	0	0
ADSS	0	0	0
AHC	0	0	0
AICART	0	0	0
AIRCr	0	0	0
ALATA_L	0	0	0
ASNS1	0	0	0
ASPCTr	0	0	0
CBPS	0	0	0
CTPS1	0	0	0
CYOR-u10m	0	0	0
CYSTGL	0	0	0
CYSTS	0	0	0
DHORD9	0	0	0
DHORTS	0	0	0
G5SADrm	0	0	0
GARFT	0	0	0
GHMT2r	0	0	0
GK1	0	0	0
GLUPRT	0	0	0
GMPS2	0	0	0
IMPC	0	0	0
IMPD	0	0	0
METAT	0	0	0
O2t	0	0	0
OMPDC	0	0	0

ORPT	0	0	0
P5CRxm	0	0	0
PGCD	0	0	0
PHETHPTOX2	0	0	0
PRAGSr	0	0	0
PRAIS	0	0	0
PRASCS	0	0	0
PRFGS	0	0	0
PROtm	0	0	0
PRPPS	0	0	0
PSERT	0	0	0
PSP_L	0	0	0
RPI	0	0	0
THBPT4ACAMDASE	0	0	0

Table D2. Results for the single-reaction host-derived enforcement analysis. Model reaction identifiers are listed. The resulting virus optima, for Chikungunya (CHIKV), Dengue (DENV) and Zika (ZIKV), are shown as a percentage of the original model (without the additional constraints imposed by the analyses), identified as normalised.

Reaction ID	**CHIKV Optima (Normalised)**	**DENV Optima (Normalised)**	**ZIKV Optima (Normalised)**
2HBO	97.69894576	87.50411496	88.88898347
2HBt2	97.69894576	87.50411496	88.88898347
2MCITt	97.69894576	87.50411496	88.88898347
3AIBt	96.86520383	100	97.45470325
3AIBTm	96.86520383	100	97.45470325
3AIBtm	96.86520383	100	97.45470325
3HBCOAHLm	96.86520383	100	97.45470325
5HLTDL	100	91.29144293	84.31838353
5HTRPVESSEC	100	91.29144293	84.31838353
ACOAD9m	96.86520383	100	97.45470325
ADEt	100	100	96.08800042
ADKd	100	100	98.19941198
ADSL1	89.17494143	71.14031648	69.28858385
ADSL2	85.93080959	71.02942248	67.37488509
ADSS	89.17494143	71.14031648	69.28858385
AHC	85.47563846	100	100
AICART	85.93080959	71.02942248	67.37488509
AIRCr	85.93080959	71.02942248	67.37488509
ARGN	100	100	97.00311453
ARGNm	100	100	97.00311453
ARGtm	100	100	97.00311453
ASPCTr	84.8214277	70.38012844	66.83842673
CAMPt	100	100	96.20591187
CBPS	84.8214277	70.38012844	66.83842673
CBPSam	100	100	96.08800042
CGMPt	100	100	97.16065675
CREATt4(2)r	100	100	97.27110861
CTPS1	73.61994513	65.09891901	61.22143988
CYSAMO	93.31242113	92.74364952	92.04750607

CYSTA	93.31242113	92.74364952	92.04750607
CYSTGL	85.47563846	81.15250586	84.63484423
CYSTS	85.47563846	81.15250586	84.63484423
DATPtn	100	100	98.19941198
DCYTt	100	100	96.27272207
DGSNt	100	100	96.08800042
DGTPtn	100	100	96.17860361
DHFR	100	100	98.52661862
DHORD9	84.8214277	70.38012844	66.83842673
DHORTS	84.8214277	70.38012844	66.83842673
DINt	100	100	97.18804875
DURIPP	100	100	98.50062324
ECOAH12m	96.86520383	100	97.45470325
GACMTRc	100	100	97.27110861
GARFT	85.93080959	71.02942248	67.37488509
GHMT2r	100	86.61097405	79.21647585
GK1	50	50	50
GLUPRT	85.93080959	71.02942248	67.37488509
GLYAMDTRc	100	100	97.27110861
GMPS2	82.69215956	70.8982818	65.53149683
GUAt	100	100	96.08800042
H2CO3D	84.8214277	70.38012844	66.83842673
H2CO3D2	84.8214277	70.38012844	66.83842673
HIBDm	96.86520383	100	97.45470325
HYPTROX	93.31242113	92.74364952	92.04750607
HYXNt	100	100	96.08800042
ILEtec	0	100	100
IMPC	85.93080959	71.02942248	67.37488509
IMPD	82.69215956	70.8982818	65.53149683
MCITS	97.69894576	87.50411496	88.88898347
MCLACCYSR	93.31242113	92.74364952	92.04750607
MCLOR	93.31242113	92.74364952	92.04750607
MERCPLACCYSt	93.31242113	92.74364952	92.04750607
METAT	85.47563846	100	100
MMMm	96.86520383	100	97.45470325
MMTSADm	96.86520383	100	97.45470325

NDPK5m	100	100	97.1449818
OBDHc	97.69894576	87.50411496	88.88898347
OCBTm	100	100	96.08800042
OIVD2m	96.86520383	100	97.45470325
OMPDC	84.8214277	70.38012844	66.83842673
ORPT	84.8214277	70.38012844	66.83842673
P5CRxm	100	100	94.78932681
PAN4PP	93.31242113	92.74364952	92.04750607
PGCD	100	100	94.89710915
PHETHPTOX2	93.50151761	100	100
PNTEH	93.31242113	92.74364952	92.04750607
PNTK	93.31242113	92.74364952	92.04750607
PPCDC	93.31242113	92.74364952	92.04750607
PPNCL3	93.31242113	92.74364952	92.04750607
PRAGSr	85.93080959	71.02942248	67.37488509
PRAIS	85.93080959	71.02942248	67.37488509
PRASCS	85.93080959	71.02942248	67.37488509
PRFGS	85.93080959	71.02942248	67.37488509
PROtm	100	100	94.78932681
PRPPS	85.37611865	70.70477546	67.10665591
PSERT	100	100	94.89710915
PSP_L	100	100	94.89710915
PUNP3	100	100	96.64096977
PUNP4	100	100	95.90899165
PUNP5	100	100	97.42933959
PUNP6	100	100	97.18804875
PYNP2r	100	100	95.64877821
RPI	85.37611865	70.70477546	67.10665591
sink_Tyr-ggn	0	0	0
TAURt4(2)r	93.31242113	92.74364952	92.04750607
THBPT4ACAMDASE	93.50151761	100	100
THRD_L	95.01363988	90.9874608	90.9874608
THYMt	100	100	98.80704485
TMDPP	100	100	98.80704485
TMDS	100	100	98.52661862
TRPHYDRO2	100	91.29144293	84.31838353

URAt	100	100	96.10807398
UREAtm	100	100	97.00311453
VALt5m	96.86520383	100	97.45470325
VALTAm	96.86520383	100	97.45470325

E RESULTS OF SINGLE-REACTION MUTATION SENSITIVITY ANALYSIS

Table E1 [Dropbox Link]. Summary of results for the host-derived enforcement analysis for different conditions: altered media conditions; virus genome point mutations; virus biomass objective function randomised stoichiometric coefficients. Reactions are selected when they are able to reduce virus optima to below 100% of the original virus optima in the original model (without additional constraints).

- https://www.dropbox.com/s/5wnt5d46p87oqqt/Appendix%20E1.xlsx?dl=0

Table E2 [Dropbox Link]. Results from the in-silico media variation sensitivity analyses to evaluate the effectiveness of the host-derived enforcement antiviral targets over a range of varied media compositions for Chikungunya (CHIKV), Dengue (DENV) and Zika (ZIKV) viruses.

- https://www.dropbox.com/s/9s50es7v2l7qxen/Appendix%20E2.xlsx?dl=0

F RESULTS OF DOUBLE-REACTION ANALYSES

Table F1 [Dropbox Link]. Double-reaction broad-targets.

- https://www.dropbox.com/s/tw0hzdl7wn06do9/Appendix%20F1.csv?dl=0

Table F2 [Dropbox Link]. Double-reaction specific-targets.

- https://www.dropbox.com/s/moqgpagr5uhn7bz/Appendix%20F2.csv?dl=0